RTO-MP-3
AC/323(SCI)TP/1

RTO-MP-3

NORTH ATLANTIC TREATY ORGANIZATION

RESEARCH AND TECHNOLOGY ORGANIZATION
BP 25, 7 RUE ANCELLE, F-92201 NEUILLY-SUR-SEINE CEDEX, FRANCE

RTO MEETING PROCEEDINGS 3

The Application of Information Technologies (Computer Science) to Mission Systems

(l'Application des technologies de l'information
(l'informatique) aux systèmes de conduite de mission)

Papers presented at the Symposium of the Systems Concepts and Integration Panel (SCI) held in Monterey, California, USA, 20-22 April 1998.

Published November 1998

Distribution and Availability on Back Cover

ORGANIZATION

North Atlantic Treaty Organization

Research and Technology Agency

RTA Headquarters: 7, rue Ancelle - 92200 Neuilly-sur-Seine, France

ST/60/4 19 August, 1998

TO: Recipients of RTO Publications

FROM: Scientific Publications Executive

SUBJECT: **RTO Technical Publications**

As you probably know, NATO formed the Research and Technology Organization (RTO) on 1 January 1998, by merging the former AGARD (Advisory Group for Aerospace Research and Development) and DRG (Defence Research Group). There is a brief description of RTO on page ii of this publication.

This new organization will continue to publish high-class technical reports, as did the constituent bodies. There will be five series of publications:

AG **AGARDographs** (**A**dvanced **G**uidance for **A**lliance **R**esearch and **D**evelopment), a successor to the former AGARD AGARDograph series of monographs, and containing material of the same long-lasting value.

MP **Meeting Proceedings**: the papers presented at non-educational meetings at which the attendance is not limited to members of RTO bodies. This will include symposia, specialists' meetings and workshops. Some of these publications will include a Technical Evaluation Report of the meeting and edited transcripts of any discussions following the presentations.

EN **Educational Notes**: the papers presented at lecture series or courses.

TR **Technical Reports**: other technical publications given a full distribution throughout the NATO nations (within any limitations due to their classification).

TM **Technical Memoranda**: other technical publications not given a full distribution, for example because they are of ephemeral value only or because the results of the study that produced them may be released only to the nations that participated in it.

The first series (AG) will continue numbering from the AGARD series of the same name, although the publications will now relate to all aspects of defence research and technology and not only aerospace as formerly. The other series will start numbering at 1, although (as in the past) the numbers may not appear consecutively because they are generally allocated about a year before the publication is expected.

All publications, like this one, will also have an 'AC/323' number printed on the cover. This is mainly for use by the NATO authorities.

Please write to me (do not telephone) if you want any further information.

G.W.Hart

The Research and Technology Organization (RTO) of NATO

RTO is the single focus in NATO for Defence Research and Technology activities. Its mission is to conduct and promote cooperative research and information exchange. The objective is to support the development and effective use of national defence research and technology and to meet the military needs of the Alliance, to maintain a technological lead, and to provide advice to NATO and national decision makers. The RTO performs its mission with the support of an extensive network of national experts. It also ensures effective coordination with other NATO bodies involved in R&T activities.

RTO reports both to the Military Committee of NATO and to the Conference of National Armament Directors. It comprises a Research and Technology Board (RTB) as the highest level of national representation and the Research and Technology Agency (RTA), a dedicated staff with its headquarters in Neuilly, near Paris, France. In order to facilitate contacts with the military users and other NATO activities, a small part of the RTA staff is located in NATO Headquarters in Brussels. The Brussels staff also coordinates RTO's cooperation with nations in Middle and Eastern Europe, to which RTO attaches particular importance especially as working together in the field of research is one of the more promising areas of initial cooperation.

The total spectrum of R&T activities is covered by 6 Panels, dealing with:

- SAS Studies, Analysis and Simulation
- SCI Systems Concepts and Integration
- SET Sensors and Electronics Technology
- IST Information Systems Technology
- AVT Applied Vehicle Technology
- HFM Human Factors and Medicine

These Panels are made up of national representatives as well as generally recognised 'world class' scientists. The Panels also provide a communication link to military users and other NATO bodies. RTO's scientific and technological work is carried out by Technical Teams, created for specific activities and with a specific duration. Such Technical Teams can organise workshops, symposia, field trials, lecture series and training courses. An important function of these Technical Teams is to ensure the continuity of the expert networks.

RTO builds upon earlier cooperation in defence research and technology as set-up under the Advisory Group for Aerospace Research and Development (AGARD) and the Defence Research Group (DRG). AGARD and the DRG share common roots in that they were both established at the initiative of Dr Theodore von Kármán, a leading aerospace scientist, who early on recognised the importance of scientific support for the Allied Armed Forces. RTO is capitalising on these common roots in order to provide the Alliance and the NATO nations with a strong scientific and technological basis that will guarantee a solid base for the future.

The content of this publication has been reproduced
directly from material supplied by RTO or the authors.

Printed on recycled paper

Published November 1998

ISBN 92-837-1006-1

Printed by Canada Communication Group Inc.
(A St. Joseph Corporation Company)
45 Sacré-Cœur Blvd., Hull (Québec), Canada K1A 0S7

Application of Information Technologies (Computer Science) to Mission Systems

(RTO MP-3)

Executive Summary

Important advances that can be expected in the coming years are:

- comprehensive information availability for a full, common tactical picture at all command levels, in the air or on the ground
- consistent knowledge availability and dissemination to command centres and mission system elements to be used by both mission system machinery and human staff
- machine capability of autonomous knowledge processing for situation assessment and decision making

The symposium dealt with the applications of advanced information technologies to mission systems and functionalities, such as

- command and control assets
- strike and defence assets (air, land, sea)
- training
- situation monitoring
- situation analysis
- planning and decision making

The main purpose of this symposium was to provide mutual education within the NATO community about the impact of information technologies, emerging or already available, on mission systems and to show the potential benefits.

On the basis of the response to the Call for Papers, the symposium was structured in the following sessions:

- Information system architecture
- Information availability about mission situation
- Knowledge availability
- Machine Capabilities of knowledge processing (methods, planning, dialogue support systems).

The papers presented generated a high level of interest. In addition to the opportunities for discussion (both formal and informal) the proceedings provide a useful reference to guide research priorities in this important area of NATO activity.

L'application des technologies de l'information (l'informatique) aux systèmes de conduite de mission

(RTO-MP-3)

Synthèse

Des avancées importantes sont prévues pour les années à venir, à savoir:

- accès sans restrictions aux informations, permettant la diffusion d'une situation tactique à tous les niveaux de commandement, en l'air comme au sol
- disponibilité permanente des informations et dissémination vers les centres de commandement et les éléments de conduite de mission, tant pour le renseignement du personnel que pour la saisie informatique
- possibilités de traitement autonome des données aux fins de l'évaluation de la situation et de la prise de décisions

Le symposium a porté sur les applications des technologies de l'information avancées aux systèmes de conduite de mission et aux fonctionalités telles que:

- moyens de commandement et contrôle
- moyens de frappe et de défense (air, terre, mer)
- entraînement
- suivi de la situation
- analyse de la situation
- planification et prise de décisions

Ce symposium a eu pour objectif de permettre des échanges au sein de la communauté de l'OTAN concernant l'impact des technologies de l'information naissantes ou déjà disponibles, sur les systèmes de conduite de mission, ainsi que leurs avantages.

Sur la base des réponses reçues à l'appel de communications, le symposium a été organisé en quatre sessions comme suit:

- architectures de systèmes d'information
- disponibilité de données sur la situation de la mission
- disponibilité de données liées à la connaissance de la situation
- capacités de traitement de ces données

Les communications ont suscité beaucoup d'intérêt. Ce compte rendu de conférence, en plus des discussions formelles et informelles qu'il peut engendrer, est une référence précieuse qui peut aider à l'orientation des recherches dans ce domaine d'activités important pour l'OTAN.

Contents

SESSION I: INFORMATION SYSTEM ARCHITECTURE
Chairman: Prof Dr-Ing R. ONKEN (GE)

SESSION II: INFORMATION AVAILABILITY ABOUT MISSION SITUATION
Chairman: Prof Dr-Ing L. CROVELLA (IT)

SESSION III: KNOWLEDGE AVAILABILITY
Chairman: Mr K. HELPS (UK)

†Paper not available at time of printing.

SESSION IVA: MACHINE CAPABILITIES
Chairman: Mr K. HELPS (UK)

SESSION IVB: PLANNING
Chairman: Prof Dr Ir A. BENOÎT (BE)

SESSION IVC: DIALOGUE SUPPORT
Chairman: Dr J. NIEMELA (US)

SESSION IVD: SYSTEMS
Chairman: Mr D. DEWEY (US)

Theme

Comprehensive information availability for a full, common, tactical picture at all command levels, in the air or on the ground, consistent knowledge availability and dissemination to command centres and mission system elements to be used by both mission system machinery and human staff, and machine capability of autonomous knowledge processing for situation assessment and decision making are all important advances that can be expected in the coming years.

This symposium will essentially deal with the applications of information technologies to mission systems, such as:

- command and control assets;
- strike and defense assets (air, land, sea); and
- training centers
- situation monitoring, making use of techniques such às:
  - machine vision;
  - speech recognition and understanding;
  - machine translation.
- situation analysis
- problem solving/planning and decision making and effecting

and will take into account techniques for knowledge acquisition (on-line and off-line learning) and data/knowledge processing/management and visualization (synthetic environment).

Thème

Parmi les avancées importantes à prévoir dans les prochaines années figurent:

- l'accès aux informations exploitables, pour l'élaboration d'une situation tactique complète et commune, à tous les niveaux de commandement, au sol et en vol;
- l'accès à des renseignements fiables et leur diffusion vers les centres de commandement et les unités de conduite de mission où il seront utilisés à la fois par des opérateurs humaines et par les systèmes de conduite de mission;
- la possibilité, par des machines, d'un traitement autonome des connaissances en vue de l'évaluation de la situation et la prise de décisions.

Ce symposium traitera essentiellement des applications des technologies de l'information aux systèmes de conduite de mission, tels que:

- moyens de commandement et de contrôle
- moyens offensifs et défensifs (terre, air, mer)
- centres d'entraînement
- centres d'élaboration de situation, faisant appel aux techniques de:
  - visualisation par la machine
  - reconnaissance et interprétation de la parole
  - traduction machine
- l'analyse de la situation
- la résolution de problèmes, la planification, la prise de décisions et sa mise en application

Le symposium prendra en considération les techniques d'acquisition des connaissances (apprentissage en ligne et autonome) et celles concernant la gestion des données et des connaissances, le traitement des informations ainsi que leur visualisation (environnement synthétique).

Systems Concepts and Integration Panel

CHAIRMAN
Dr E STEAR
The Boeing Company
PO Box 3999
Mail Stop 85-93
SEATTLE, WA 98124-2499
UNITED STATES

DEPUTY CHAIRMAN
Prof L M B da COSTA CAMPOS
Instituto Superior Tecnico
Torre-6° Piso
Avenida Rovisco Pais
1096 LISBON CODEX, PORTUGAL

TECHNICAL PROGRAMME COMMITTEE

CO-CHAIRMEN:	Prof Dr Ir A BENOÎT	BE
	Prof Dr-Ing R ONKEN	GE
MEMBERS:	Dr-Ing L CROVELLA	IT
	Mr K HELPS	UK
	Mr D DEWEY	US
	Mr L HOLCOMB	US
	Dr J NIEMELA	US

PANEL EXECUTIVE

From Europe:

RTA-OTAN
LTC T ROBERTS, USA
SCI Executive
BP 25, 7 Rue Ancelle
F-92201 NEUILLY-SUR-SEINE CEDEX,
FRANCE

Telephone: 33-1-5561 2270/82 - Telefax: 33-1-5561 2298/99

From the USA or CANADA:

RTA-NATO
Attention: SCI Executive
PSC 116
APO AE 09777

HOST NATION LOCAL COORDINATOR

Mr J K RAMAGE
Chief, Flight Control Development Branch
WL/FIGS, Bldg 146
2210 Eighth St, Suite 11
WRIGHT-PATTERSON AFB, OH 45433-7521
Tel: (1) 513 937 3047
Fax: (1) 513 656 7505

ACKNOWLEDGEMENTS/REMERCIEMENTS

The Panel wishes to express its thanks to the United States RTB members to RTA for the invitation to hold this Symposium in Monterey and for the facilities and personnel which made the Symposium possible.

Le Panel tient à remercier les membres du RTB des Etats-Unis auprès de la RTA de leur invitation à tenir cette réunion à Monterey, ainsi que pour les installations et le personnel mis à sa disposition.

TECHNICAL EVALUATION REPORT

Larry Ott
1325 Mill Creek Rd.
Southampton, Pa. 18966
United States

Introduction

The first Systems Concepts and Integration Panel (SCI) Symposium was held in Monterey CA. USA, from 20-23 April 1998. The Symposium titled "The Application of Information Technologies (Computer Science) to Mission Systems" addressed what is known as the "soft technologies". The purpose of the symposium was to deal with applications of Information Technologies in mission systems like command and control assets, strike and defense assets, and training centers. Functionalities to be addressed included:

- Situation monitoring, making use of techniques such as
  - Machine vision
  - Speech recognition and understanding
  - Machine translation
- Situation analysis
- Problem solving/Planning and Decision Making and
- Effecting

taking into account techniques for knowledge acquisition (on-line and off-line learning), data/knowledge processing/management and visualization (synthetic environment).

The program under the overall chairmanship of Prof. Dr-Ing Helmut Sorg of Germany was divided into seven sessions containing a total of 22 papers. The papers included technologies such as data fusion, genetic-based algorithms, neural networks, machine intelligence, crew assistants, information processing architectures, and the application of these technologies to various platforms. The symposium attracted some 90 participants. The smaller than usual attendance could perhaps be attributed to this being the first symposium since the transition from Advisory Group for Aerospace Research & Development (AGARD) to the Research and Technology Organization (RTO) and also due to the subject which is just beginning to make its way from a laboratory science to the applications field.

Summary

This was a very broad coverage of a relatively new science. The approach the symposium took from presenting tutorial type papers to actual applications provided the attendees with an overall appreciation of the topic. The papers did meet the symposium's objective stated in the Introduction. Being so broad however, it was difficult to get an in-depth treatment of any one subject. Some attendees liked the broad scope while others did not. Following each paper there was for the most part a good dialogue between the presenter and the attendees. The session chairmen did an excellent job in keeping the papers on schedule. The papers followed in a logical manner although some of the papers would have benefited from being placed elsewhere. The symposium not only presented innovative solutions in an attempt to solve the information warfare problem but also provided examples of these new technologies being made available to the operational forces. General themes running through the sessions included:

- The need for affordability and automation
- Current computer technologies are inadequate to meet the affordability and automation needs
- The new soft computer technologies when used in conjunction with conventional processing technologies have promise in meeting the goals of affordability and automation
- The information technologies are for the most part still in an embryonic state requiring considerable technical advances before they gain the confidence of the warfighter, although some of the papers showed significant accomplishments
- A limited capability of some information technologies have been transitioned to the warfighter with some success
- There is considerable effort ongoing to provide an automated "crew assistant" for the pilot and crew

The attendees rated the symposium on average "important" with 25% rating it as "significant". Some of the comments included, "liked the broad aspect of the program", "not technical enough, lack of focus", "able to meet and discuss technologies/applications with others in the field".

Recommendations

Since the symposium covered a broad array of topics and some of them relativity new, it would have been helpful for the attendees to have gotten an overview of the symposium, the theme, and the approach at the beginning of the meeting. At the beginning of each session, this approach could have continued by having the theme and focus of the session described and how it fitted in with the overview. As mentioned earlier, some of the papers could have been fitted in different sessions to provide a better flow. Some of the past symposiums have had a panel discussion at the end to discuss the topic that has proved helpful in providing further focus on the subject.

For future symposiums, it is recommended that the author spell out acronyms in his/her presentations. The audience in many cases is unaware as to the meaning of the acronym.

Future symposiums on this topic may want to focus on either technology developments or the applications of these technologies. This may address the focus concerns of some of the attendees. However as the first symposium on the topic, this sets the stage for further discussions on technologies which could have a significant impact on warfighting.

Technical Content

The Acting Superintendent of the Naval Postgraduate School, Capt. J. Burin welcomed the symposium attendants. He pointed out that it was appropriate that the symposium should be held at the Postgraduate School in that students from 35 countries were attending classes in Monterey. Captain Burin described his past assignments working in support of NATO and emphasized the importance of technology for the warfighter especially information technology.

Mr. Jim Ramage, the acting chairman for the SCI panel, described the structure for the RTO. It essentially has three layers, the policy makers, the panel members, and the technical project teams. There are six panels: Applied Vehicle Technology; Human Factors and Medicine; Information Systems Technology; Studies, Analysis, and Simulation; Systems Concepts and Integration and Sensors and Electronics. Each panel will have 50 members with each nation having 3 members and 1 vote per nation. The panels will now include all vehicles not just aerospace as the former AGARD panels did.

Keynote Address

Mr. Robert Kahn, of the Corporation for National Research Initiatives gave the opening address. Mr. Kahn described an infrastructure that is open in its architecture and which supports a large and extensible class of distributed digital information services. In such a system, information is stored, accessed, disseminated and managed. Key attributes of such a system include structured information (digital objects), unique and resolvable identifiers, repositories to store the digital objects, stated operations for each object all integrated in an open architecture system. He described the architecture of such a system, which included a search mechanism, user interface, authority systems, registers serving as repositories and an interactive rapid look mechanism. For the repositories he took a "nesting approach" which included a core, structure, content, and aggregation and deaggregation.

Mr. Kahn described the operation of the system and defined in some detail its various parts. He defined the digital object as a data structure whose principal components are digital material or data, plus a unique identifier for this material, called a handle. A digital object could be changed (mutable), or not changed (immutable) after it is stored. He also described how to access the stored data through the Repository Access Protocol (RAP). He went into some detail on the Handle Server Infrastructure including imposing semantics on handles. The handle is assumed to have two logical components, a local naming authority name, and an identifier unique to that naming authority. He made the point that particular naming authorities could follow their own conventions for assigning semantic or non-semantic strings for their objects.

In summary, Mr. Kahn, described a system for naming, identifying and or invoking digital objets in a system of distributed repositories that provides great flexibility and is well-suited to a national-level enterprise. It allows the possibility of locating digital objects without making any presumptions about the objects locations. Furthermore, it allows the local user to employ its own federated system while being part of the open architecture system. Users of such a system include the Library of Congress, the U.S. Information Agency and the Defense Technical Information Center.

Session 1: Information Systems Architecture

The papers of this session described the role of information for the warfighter, a new information processing architecture and a paradigm shift in processing information. Although these papers were diverse, they provided an introduction of the topics that were to follow. What can be taken from this session is that information will be paramount for the warfighter and technology is advancing to provide this information to the various elements of the command structure. However progress is needed to provide the right information at the right time to the right element of the battle group. This is particularly true in providing more automation in processing sensor information. To do these functions efficiently and effectively, a new way of thinking in processing information may be required. From the papers presented, it appears as information can be transmitted effectively, but we are only scratching the surface in developing the new technologies to process this data using more automated techniques.

The first paper, "Evolution to Integrated Command and Control" by DeRosa and Woodall of MITRE Corp. U.S. emphasized the importance of collecting and distributing information across the battlefield while denying the enemy to do the same. The key in accomplishing this is to move from a number of disparate command and control systems to an integrated system. The authors proposed a C2 operational vision of the future, which included a common tactical picture for the various warfighters including the combatants. The elements of the operational vision include a global communications grid, tailored situational awareness, and dynamic planning and execution. The global communications grid consists of interconnected networks operating as one global, high availability secure network. Tailored situational awareness provides a common view of the battlespace for users ranging from the highest levels of command to individual users. The dynamic planning and execution element provides tools to carry out offensive and defensive planning and replanning, weapons control, and combat support. The authors pointed out that the system architecture is a key item in this C2 vision. The architecture has to consider the operational, technical, and systems views. Software technologies, such as CORBA, JAVA, and the Web are emerging to make the concept viable. The authors envision a "total information wall" located at the Operation Center, which will provide complete battlespace awareness. An evolutionary process was described to expedite the transition of this concept to the warfighter.

The second paper " Information Processing Architecture for Mission Performance of Autonomous Systems Capable of Dynamic Vision" by Dickmanns and Furst of Germany discussed the automatic processing of imagery data. The authors proposed a 4-D approach that requires an interpretation of image sequences both in 3-D space and over time simultaneously. The system should have a wide field-of –view with a central area of high spatial resolution. The large field-of-view will allow the tracking of several objects simultaneously. The paper lists 13 assumptions underlying the 4-D approach including efficient interpretation of sensor signals and efficient computing. In this approach, there are three main activities running in parallel: detection of objects, tracking of objects and state estimation and learning from observation. Since situations present different interpretations, the authors introduced a hierarchy of goal functions and value systems for creating the capabilities of complex decision making in an autonomous system. An overall system integration concept with five levels was presented. The highest level, "mental processing" is just being addressed. An initial, rudimentary experiment has been performed using different focal length cameras on a helicopter platform that showed the feasibility of the concept. Applications for this technology, which requires further advances in computer technology, include Unmanned Combat Air Vehicles, Nap-of –the earth flights of helicopters and any other application, which could reduce pilot workload.

The third paper in the session, "Advances in Soft-Computing Technologies and Applications in Mission Systems" by Krogman of Germany discussed a paradigm shift from how processing is done today and how it may be done in the future. According to the author, there is a shift from the conventional computing techniques, including symbolic artificial intelligence and knowledge techniques to the soft computing technologies. This is based on modeling the unconscious, cognitive, and reflexive function of the brain. The introduction of computational and machine intelligence (CMI) techniques will allow this to happen. Included under the title of CMI are technologies such as neural networks, genetic algorithms, and fuzzy logic. These technologies will complement "standard technologies" such as artificial intelligence, decision theory, and computer science. In the old view, computation was based on sequential computing, artificial intelligence networks and learning by rules. In the new view, we will have dynamic processing, artificial intelligence neural networks and learning by example. We will be shifting from a crisp to fuzzy representation using technologies such as genetic algorithms. Complex, highly integrated systems, requiring high degrees of automation and intelligence will result from the addition of these new "soft computing" technologies.

Session II- Information Availability about Mission Situation

The authors in this session discussed the fusion of on-board sensors and also pre-flight stored imagery data. Also presented was a paper on how on-board fused sensor data can be enhanced using fuzzy logic. The second paper in this session was not presented although the paper was furnished. Fusion is a topic that makes intuitive sense in that you are making the task of the pilot or crewmember easier by integrating information from various sources whether on or off- board the platform. A pre-loaded three-dimensional map with real-time sensor data superimposed is a particular attractive approach. However, to transition this capability from the laboratory to the warfighter has become a major challenge. The papers in this session provide efforts to overcome this challenge. One of the papers presents a testbed architecture that can be used to evaluate new techniques prior to implementing them in actual military hardware.

The first paper "Image data fusion for enhanced situational awareness" by Dohler, Hecker, and Rodloff of Germany discussed the fusion of image data from onboard multispectral sensors (EVS-Enhanced Vision System) with synthetic vision, i.e. pre-stored image data along with on-board navigation data. The papers states that these are sometimes considered competing technologies but are really complementary. Each has its advantages and disadvantages, but if integrated can provide the pilot with a definite advantage almost to the point of having an assistant. A block diagram of the system was shown which included elements of both the on-board and synthetic subsystems. Key elements of the system include the vision processing and vision fusion components. Based on both civil and military requirements, the requirements for imaging sensors were provided. A table with the characteristics of potential EVS sensors including Ultra-Violet, infrared, Millimeter Wave and Passive Millimeter Wave was given. A discussion of the viability of the sensors regarding weather, detectability, and image generation ensued with a summary of the "good and bad " features of each of the sensors. A particular sensor, active mm-wave radar, was used as the main sensor for EVS research. This sensor, which is all weather, has the advantage of having a longer range than a hostile ESM receiver. This sensor output overlaid with a terrain data or fused in time or space offers promise as an all-weather imaging sensor.

The second paper 'Theater and Multi-Sensor Large Data Intelligent Handling, Storage, and Presentation System" by Mura and Cappellanti of Italy was not given although the paper was provided. This paper describes a project for the management of a large amount of data. Research issues include systems requirements definition, multi-sensor and geographical data fusion, large data base storage and data display.

The third paper, "Software Testbed for Sensor Fusion Using Fuzzy Logic" by Stubberud and Lugo of Orinicon Corp, U.S. discussed the use of a software testbed to assess and evaluate algorithms using new techniques such as fuzzy logic and neural networks to improve the data fusion techniques. In order for this to work, all elements of the data fusion system must be able to share information with each of the other elements. The testbed is somewhat unique in that besides the sensor information it also has target databases and environmental maps all communicating with each other. The sensor logic subsystem has five subcomponents, including two sets of error models. Regarding the data association module, not only are standard association techniques used but also fuzzy association algorithms. Also for the data fusion, augmented and intelligent algorithms are used in addition to the standard data fusion algorithms. This testbed was designed to be modular since the fusion area is an embryonic technology with no sure path to the final system.

Session III- Knowledge Availability

This session began with a tutorial on the potential of soft-computing methods for mission systems and was followed by papers on the application of three soft-computing technologies, fuzzy logic, real- time object structuring and real-time simulation. The tutorial followed the earlier paper by Krogman on the technology of softcomputing and provides some additional detail. The paper on fuzzy logic described how fuzzy logic rules could be learned from training data. The third paper presented the thesis that defense systems are becoming so complex and costly that new approaches must be used if these new capabilities are to be developed. Again these papers showed that these technologies are still in an embryonic state.

The first paper in the session, "The potential of soft-computing methods for mission systems: A tutorial," was presented by van der Wal of TNO-FEL. Physics and Electronics Laboratory, The Netherlands. The author presented an overview of the principles and basic concepts of four basic softcomputing techniques; fuzzy systems, neural networks, genetic algorithms and ordinal optimization from the point of view of their potential use in military mission systems particularly simplifying information coming to the operator. Since these mission systems undergo rapid change and deal with uncertainty, soft computing technologies provide a good match. They are easy to apply, robust, human-friendly, can handle ambiguous and sometimes conflicting information, can learn and do human-like inference. However, a general theory encompassing all softcomputing methodologies is still lacking. The author also described the integration of softcomputing techniques with classical methods. He provided a hierarchy of modeling techniques stating that all methods could be combined but a model higher in the hierarchy should be used first. The softcomputing techniques are still in their infancy but the author believes that they are key for the success of future mission systems.

The second paper "Learning Fuzzy Rules from Data" by Hammell, U.S Army and Sudkamp, Wright State University, U.S. described a hierarchical architecture for fuzzy modeling and inference that learns the underlying fuzzy rules from training data. The authors show that a combination of fuzzy associated memory (FAM) and the error function (EFAM) provide the best data, better than the FAM alone. Impact of granularity (increasing the number of fuzzy sets) and dimensionality (adding additional input variables) was also discussed. Increasing granularity and dimensionality requires more training data. Regarding run-time, granularity has no effect while dimensionality could have an exponential effect on the time needed to evaluate the rules. The authors described ongoing research combining domain expertise and learning algorithms.

The next paper in the session "Real-Time Object Structuring and Real-Time Simulation for Future Defense System Engineering" by Kim and Stubberaman of the University of California, U.S. cites the two-aforementioned technologies, which the authors believe, are among the most important to support future engineering of advanced defense systems. The design complexity of large-scale command and control systems are requiring new techniques in order to control costs and improve reliability. A major goal is to leverage commercial developments. Real-time object structuring provides the production of systems designs, which are easy to understand and modify, and also allows the reuse of modules tested in earlier applications. Real-time simulation is an advanced mode of simulation in which the simulation objects are designed to show the same timing behavior that the simulation targets do. The authors discussed a real time object-structuring scheme called the "time-triggered, message-triggered object (TMO) structuring scheme and applied this to an anti-missile defense scenario. The advantages of this TMO based design and simulation were provided.

Session IVA- Machine Capabilities of Knowledge Processing –Methods

The two papers in this session discuss an Integrated Processing System and the application of genetic algorithms to military systems. In order to handle the software technologies discussed in this symposium, a systems architecture that is open and COTS based and can be used across a number of different platforms is required. The paper in this session described such an architecture. The second paper discussed the potential application of genetic algorithms to a number of military applications.

The first paper, 'Morpho Systems" by Singh, Lee, Lu, Kurdahi, Bagherzadeh, of the University of California, U.S., Heaton of Obsidian Technology, U.S. and Filho of the Federal University of Rio de Janeiro, Brazil, presented the design and development of an integrated re-configurable processing architecture. A SIMD (Single Instruction Multiple Data) Architecture was used. Two chips have been developed for use in the architecture that consists of 8 by 8 configuration. The key to the architecture is that it is reconfigurable.

The second paper, "Using Genetic-Based Algorithms to Mission Systems Application" by Krouel and Williams of the UK, discussed how mission systems are becoming much more complex and why genetic-based algorithms should play a significant role in solving these problems. The advantages of these types of algorithms include their ability to operate in a noisy, non-static environment, can solve complex problems without using significant computer overhead, can provide multiple solutions and are easily implemented. Relevant application types include planning systems, both strategic and tactical, scheduling, and autonomy. Mission system applications include satellite frequency planning, convoy movement, route planning, weapon target assignment, and battlefield simulation. For each of these applications, the authors described genetic algorithms that have been or could be used. The authors do point out that genetic algorithms are not a panacea and should not be used if sound analytical methods exist. However, for a class of problems that are noisy, uncertain, and dynamic, genetic algorithms should have a role in helping to provide a solution.

Session IVB-Planning

The papers in this session discuss the application of distributed planning systems for both commercial and military applications. In most cases, a combination of mathematical and domain expertise is used for the applications cited. Technologies such as decision trees, blackboard architectures and expert systems are used. The applications, classified as the real-time interaction of many bodies, include airport traffic management, ballistic missile defense and destruction of many targets by multiple aircraft.

The first paper of the session, "Airport Traffic Management based on Distributed Planning", by Bohme of the Institute of Flight Guidance, Germany, discussed methods and algorithms for the integration of distributed planners for arrival, departure and ground management of airport traffic. The reason for doing this is to share resources and to ensure efficient management of the traffic. The key is the interdependence and the easy update of the planning systems. A blackboard architecture including a data base management system was used to integrate the planning units. The author pointed out that in the paper the static case was described i.e. the situation at a certain time. However in the real world, with information constantly changing, plans must be adapted constantly. Also since humans are involved, there must be some stability to the plans. In response to a question, the author had given equal importance to both arriving and departing flights. The model would have to be updated to give more importance to one or the other.

The next paper, "Optimal Decision-Making and Battle Management" by Trivizas of Greece presented the concept of Dynamic-Decision making using the space-based battle management as an example. The scenario was described, the invasion of nuclear warheads targeting cities and military bases in the U.S. The objective of the defending system was to eliminate the warheads particularly those that would destroy high value targets. The battle is described first from a static and then from a dynamic view. A decision tree is the central concept of the real-time decision making, containing the sequence of optical tactical decisions. The optimal decision sequence may only be found by enumeration of all possible tree outcomes. To reduce decision times, heuristic approaches are coupled with that of the decision tree. The author concludes the paper with an example of Dynamic Battle Management using the techniques outlined in the paper.

The final paper in the session,"On Vehicle Allocation to Targets in Mission Planning" by Choenni of the National Aerospace Laboratory, The Netherlands, presented a formalism that would encompass a number of diverse military mission planning activities. The example used was the allocation of targets to aircraft, located in different geographical areas, in the least costly manner. The location of the aircraft was known but the location of the targets was not. Since target location was updated continuously, the plan had to be easily updated. A mathematical model, based on a cost function, was developed for the application cited above and this was combined with domain knowledge provided by an expert or database. The latter was done to reduce the complexity of the model. A list of targets that is feasible for each aircraft is generated and then for each aircraft a list of targets is selected. A strategy for updating plans with minimum effort was also given. Parts of the approach have been implemented successfully but memory problems were encountered during the process of selecting targets since the decision tree became too large.

Session IVC- Dialogue Support

The title of this session is certainly appropriate for the papers presented. Each of the papers described technology that is helping the warfighter. What is remarkable is that that these technologies were fielded early to provide immediate help. Normally one thinks of it taking up to 10 years to field a new system. The papers in this session proved this thinking to be out-of-date. What was done, instead of waiting until the technology was perfected, the technologies took what was available and provided some capability to the warfighter. In many instances this "limited" capability was sufficient to provide significant capability.

The first paper," High-Mobility Machine Translation for a Battlefield Environment" by Holland and Schlisiger of the Army Research Lab, U.S. described a lightweight (28 lbs.), COTS developed system which automatically translates languages into English. The translation is rough but users have stated that it is 80% accurate. It has been used in Bosnia for over a year. Documents are first scanned which provides a bitmap image. An optical character reader (OCR) recognizes characters in the image and generates text files. A translator then converts the text to English. The system has allowed non-linguists and non-experts to evaluate foreign language documents in a field environment. It also has assisted linguists to speed up translation of documents. Improvements are needed particularly in the development of the OCR to handle lower quality or dirty paper.

The second paper, "C4I for the Warrior: Supporting Operation Joint Endeavor" by Lepanto and Serben of the Draper Laboratory, U.S. presented a COTS based communication system that was demonstrated in Bosnia. The architecture consists of a communications and networks infrastructure, with information servers and information management software. The communications infrastructure comprises the Joint Broadcast Service (JBS) and the Very Small Aperture Terminal (VSAT) Network. The VSAT network provides full duplex satellite communications capability between sites equipped with VSAT terminals and receivers. The system has demonstrated multimedia transfer using asynchronous transfer mode (ATM). The system can access a wide variety of data including that from aircraft, both manned and unmanned and information sources such as the Defense Intelligence Agency and National Imagery and Mapping agency. The system was first tested in the United States before being employed in Europe. Lessons learned included the importance of configuration management, user training and interaction with the end user. Although the system has the capability in near real-time (5-30 sec.delay), to transmit a large amount of imagery and data, it underscores the importance of transmitting only the information needed. Otherwise the warfighter could be inundated with information he does not need.

The last paper in the session, " Introducing Machine Intelligence and Autonomy into Satellite Communications Systems" by Krouwel of DERA, U.K. examined the U.K. Military Satellite System, recognized shortfalls and provided solutions. The two areas cited by the author that should be addressed include the provision that the system is always available and operable and that the user community's needs are met. The first need would be met with the introduction of automation. If a component failed, this failure would be automatically detected, and a replacement component would be inserted via reconfiguration. A problem does arise with legacy equipment that may not have the capabilities of newer equipment. The author proposed a system to provide computer control and monitoring of older systems. Regarding meeting the users' needs, two approaches are presented. The first, the Network Planning Tool, takes the users requirements, assesses the availability of the various components and automatically decides how to route the information. The software is very rapid, taking less than a second to produce plans for a set of over 150 user requirements. The second, Response to Stress, deals with equipment failures, operator error or interference. A Stress Recovery Assistant (SRA) is used to provide advice to operators at all stages of the response to the stress process. The architecture consists of a series of information 'noticeboards' and specialized agents. The SRA consists of a mixture of AI and non-AI agents. The SRA performs diagnosis and serves as a recovery agent.

Session IVD- Systems

This session continued the theme of the previous session; machine assistance for the operator. The six papers discussed different approaches but all have the same goal in mind. The 'crew assistant' concept has been around for some time and some ambitious programs have been undertaken in this area. To date there has not been too much success as far as fielding systems. However as described in this session, the projects are being undertaken in smaller steps and are being tested as they go along. Laboratory testing is followed by flight tests. With this approach and focused applications, these programs could be successful. Certainly the assistance is needed for the crew and if unmanned air combat vehicles are to become viable, this type of technology is required. It was not clear from these crew assistant papers how much of the new soft technologies are being used; perhaps a better marriage could be undertaken between the technologist and user. The last paper in the session did incorporate the new technologies using both artificial neural networks and genetic algorithms. In well-conducted experiments, new ground appears to have been broken in the automatic target classification field.

The first paper, " The Cognitive Assistant and its Contribution to Effective Man/Machine Interaction" by Flemisch and Onken of University of Armed Forces, Germany, presented the concept of the machine working in partnership with the operator in performing the mission. In order to be effective, the machine must have a full picture of the flight situation and focus the crew on the most urgent task. An initial cognitive system was developed for IFR aviation and based on that experience a crew assistant was developed for military transport aircraft. A main feature of the system is a module, which takes inputs from the sensors, and situation modules, compares this information with the flight plan and identifies discrepancies as errors or intents. The system has been tested in a simulator and flight tests are scheduled for 2000. The system offers a feature called the Dialogue Manager (DM) which expedites communications between the pilot and the machine. Speech or textual interface, touch screen, prioritization of messages, and advice, handles information overload by the machine of a changing situation which the pilot can accept or reject. In response to a question, the author stated that databases are very important for this concept.

The second paper, " Machine Intelligence Applied to Future Autonomous Tactical Systems" by Krogman of Bodenseewerk Geratechnik, Germany, discussed some of the enabling technologies that will allow the applications cited in this session. The crux of the paper was that affordability for defense systems is becoming paramount and that means there will be more emphasis on autonomous/unmanned systems. These systems are characterized as functioning independently over an extended period of time. They must have the capability to learn, solve problems, resolve conflicts and be capable of decision making. They will concentrate on performing tasks within the environment in which they are interacting and adapting to the changes in the environment to meet the goals they have been given. For this to take place there must be a paradigm shift to brain-like information processing structures. Emphasis will focus on machine intelligence rather than computational efficiency. Computational and Machine Intelligence technologies, addressed by Krogman in an earlier paper, such as fuzzy logic, neural networks, and evolutionary algorithms will be increasingly important. A brief description as to how these technologies could be used in an automated system was given. Two different architecture approaches could be used, either a top-down or bottom-up. For the top-down or hierarchical approach, the system is structured in a series of levels or layers following the concept of increasing precision with decreasing intelligence when going from top to bottom. For the bottom-up or behaviorist architecture, so-called agents are implemented with the simplest action and behavior patterns possible so that the resulting emergent system behavior corresponds to the desired objective. The latter is the preferred architecture to provide the type of automation cited in this paper. A question was asked as to how quickly could this technology get to the warfighter. The answer was that it was somewhat dependent on the confidence in these new technologies such as fuzzy logic.

The next paper in the session, "Crew Assistance for Tactical Missions in Simulator and Flight Trials" by Schulte and Klockner of ESG, Germany, described an approach to crew assistance to low-level flight missions. This paper continued the discussion of the first paper in this session and applies the technology to a tactical mission vice a transport. The paper focuses on two aspects: a Tactical Mission Management System and the Crew Assistant for Military Aircraft. The Tactical Mission Management System consists of four elements: Tactical Display, Tactical Situation

Interpreter, Low-level Flight Planner, and a Primary Flight Display. Map data obtained from DTED and DFAD along with threat data obtained via the Tactical Situation Interpreter is shown on the tactical display. Given this information, the low-level flight planner determines the optimum route. An out-the-window three-dimensional view is presented on the primary display. The Tactical Mission Management System was flight tested on a Dornier 128 aircraft. The tests were successful in that the crew was able to perform automated flight replanning. The synthetic vision format needed improvement. The second thrust of the paper, the Crew Assistant, was tested in a simulator. Based on values from –3 to +3, the pilots gave the Crew Assistant a value that ranged mostly from 2 to 3. The questions ranged from the evaluation of situation awareness to assistance quality. Flight tests for the crew associate are scheduled for the early 2000.

The fourth paper in the session, " Information Decision or Action – the Role of IT in Fast Jet Mission Systems" by Semple of British Aerospace Military Aircraft & Aerostructures, UK, focused on the decomposition of the mission into information, decision and action. The author states that by doing this the most mission processing can be reduced to deterministic code, leaving the intelligent processing within the capacity of the crew. The challenge is to be able to ascertain in real time how information is to be decomposed and what is appropriate for the machine and for the crew. The author then described enabling technologies such as data fusion, tactical situation assessment, and tactical decision aids and pointed out that computer capabilities make these technologies feasible. Tactical situation assessment is further divided into defining the situation, interpreting the situation and monitoring the pilot if he should make an error. Regarding whether decision aids should be advisory or informative, it is dependent somewhat on the experience of the crew. The more experienced, the less of a need for advice. Two other areas discussed where computer technology can now be applied include mission re-planning and information display management.

The fifth paper in the session, "Knowledge Based Decision Support Technology Demonstrator Program for a Maritime Air Mission System" by Howells and Davies of DERA, UK and Macauley and Zancanato of MoD ,UK, described Knowledge Based Decision Support systems for Maritime Support Systems. Applications include Anti-Submarine Warfare (ASW), Anti-Surface Warfare (ASUW) and Airborne Early Warning (AEW). The approach for the ASW and ASUW mission is different than that for the AEW mission although both approaches use knowledge based systems. For the ASW/ASUW mission consists of a computer simulation environment, a data fusion capability, decision support workstation and real-time multi-agent software toolkit. The architecture that will be used for a flight demonstration program will be reconfigurable. There would be a capability to fuse ESM, radar, and acoustics, and provide general assistance to the TACCO. For the AEW application, a system is being developed that will use the computer and crew in a cooperative fashion. A series of interviews have been held with subject experts to elicit knowledge. Further knowledge acquisition is being done using relevant tool kits such as PCPACK. The architecture is being designed to ensure the reuse of components, and the ability to grow the system without reengineering the system.

The final paper in the session was "Applications of Artificial Neural Networks and Genetic Algorithms for Electromagnetic Target Classification" by Turhan-Sayan, Ince, and Leblebicioglu of Middle East Technical University, Turkey. This paper addressed the classification of targets using the scattered radar field data. This data is dependent on polarization and aspect angle of transmitted and received radar signals as well as the frequency band. A target classifier must be able to handle the data in real time. As the title of the paper indicates, two approaches were investigated as target classifiers. The first made use of artificial neural networks (ANN) together with time-frequency signal representation. Artificial neural networks have been used in these types of applications because of their ability to learn and generalize and their capacity for massive parallel processing. A feature extraction capability was used to assist in the classification. Simulation was used to test the classifier and using five targets. Training was included in the simulation. The classifier had a correct classification rate of 93% and the CPU time was a fraction of a second. The second approach used genetic algorithms for the passive recognition phase for the design of characteristic pulse signals for each candidate target. Tests were run using genetic algorithms in K-pulse shaping for a perfectly conducting cylinder thin wire. The error in using the genetic algorithms was less than 0.1%. Both of these approaches offer considerable for automatic target classification in real time.

Evolution To Integrated Command and Control

Dr. Joseph K. DeRosa
Mr. David Woodall
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730 USA

SUMMARY

Significant improvements in military capabilities continue to be enabled by technological advances in offensive and defensive weapons, and improvements in communications, sensing, and information technology. Many of these capabilities and technologies are openly available to potential adversaries through the international marketplace as well as by internal development and third-party acquisitions. A key discriminator for measuring the effectiveness of Allied Forces in joint and combined operations is the ability to collect and distribute an uninterrupted flow of information across the battlefield, while denying the enemy the ability to do the same. Ensuring military superiority for a variety of missions in future operations will be strongly dependent on our ability to integrate these new capabilities into an overall system and operational concept. Thus, disparate command and control systems must come together as a single overall system executing a coordinated operational concept. Technologies and methods are emerging to make the concept of a single, integrated command and control system a reality.

ACRONYMS

C2	Command and Control
C2STA	C2 System Target Architecture
C3I	Command, Control, Communications & Intelligence
C4I	Command, Control, Communications, Computers & Intelligence
CONOPS	Concept of Operations
COTS	Commercial Of-The-Shelf
CVBS	Common View Of The Battlespace
DII-COE	Defense Information Infrastructure-Common Operating Environment
EFX	Expeditionary Force Experiment
FOL	Forward Operating Locations
IDL	Interface description language (IDL)
ISR	Intelligence, Surveillance & Reconnaissance
IT	Information Technology
JCS	Joint Chiefs of Staff
ORB	Object Request Broker
RMI	Remote Method Invocation
RSTA	Reconnaissance, surveillance and target acquisition

INTRODUCTION-CONTEXT

Weapons, command and control (C2), and reconnaissance, surveillance, and target acquisition (RSTA) sensor-platform technologies are developing rapidly. The proliferation of systems resulting from these developments as well as the availability of sophisticated commercial technologies present new challenges. Threats and force structures are also changing. Therefore, a key discriminator in ensuring superiority over any opponent will be the ability to integrate the command and control (C2) of these advanced systems. Ideally the C2 would be managed in a cohesive manner and behave as much like a single, integrated system as does any traditional weapon platform, e.g. a fighter aircraft. However, due to the large investment in the current installed base of the separate C2 systems in the NATO Alliance, that is not possible in the near term. What is possible with emergent information technology (IT) is to both build new systems and migrate legacy systems into what would appear to be a single integrated C2 system.

A number of important trends listed in Table 1 have emerged during the 1990s and their implications for mission management integration were discussed in an earlier paper [Reference 1]. Many of the raw capabilities listed are commercially available to anyone, significantly increasing the threat to NATO's military and civilian infrastructures. Supporting smaller scale contingencies while faced with increased operations in space by a number of countries requires an increased operational tempo as well as a dispersing of forces.

The trend toward higher lethality reflects the results of recent technological advances in reducing the "signature" of strike aircraft and missiles and fielding new guidance capabilities to improve the accuracy of delivering munitions. This trend is continuing and is being complemented by improving remote sensing. In addition, "information warfare" (IW) capabilities are advancing rapidly in the "hacker" and other communities, which may present major threats to our systems in the 21st century. Many of these advances are both driving us to and enabling the increased use of unmanned weapons and RSTA air-vehicles. Improvements in the miniaturization of components and systems will continue to be a major enabler of new force, C2, and RSTA capabilities. Advances in information technology and modeling and simulation (M&S) are also expected to have a major impact on future capabilities.

Some Important Trends

1. Supporting multiple smaller scale contingencies
2. Downsizing military forces
3. Higher lethality (and cost) weapons
4. Proliferation of advanced weapons and supporting capabilities to a wide variety of potential adversaries
5. Changing threats and capabilities
6. Increased use of space assets
7. Improving UAV capabilities - unmanned weapons and RSTA
8. Information Warfare (IW)
9. Advances in information technology
10. Miniaturization of components
11. Modeling and simulation for operations and training

Table 1

The challenges posed by these trends require innovation in command and control.

The development and operation of an integrated C2 system will allow the Warfighter to seamlessly and collaboratively plan for missions from the simple to complex composite combat missions by scaling C4I to the needs at hand. Ideally the Warfighter would have available a global communications grid to enable robust global operations and tailored situation awareness to gain a common view of the battle space (CVBS). They must also have available the means to dynamically plan and execute missions in collaboration with Alliance and

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

coalition partners. The challenges posed by the important trends in Table 1 can then be met.

This paper discusses the vision of building an integrated command and control system. It defines the characteristics of such a system and suggests a path for Alliance investments in technologies that will help migrate from the current state of C2 to that vision. It proposes strategies for acquisition, technology insertion, and development of new operational concepts.

C2 Operational Vision

Joint Vision 2010, issued by Gen. John M. Shalikashvili, Chairman of the Joint Chiefs of Staff (JCS) of the United States, provides overall direction for American military services into the next century. Joint Vision 2010 outlines several basic trends to consider when planning how to meet the challenges facing the nation's armed forces. These trends follow closely those presented in Table 1: (1) technologies in the military arsenal are widely available and affordable; (2) contingencies are more complex and numerous than ever before; (3) the military must be prepared for short-notice actions; and (4) continued deficit reduction efforts will result in greatly reduced operating budgets.

Many of the same trends that are driving the development of future C4I and weapons within the NATO community are also driving future U.S. national systems. All these future C4I and weapons systems must support the needs of tomorrow's WarFighters. To do this, they must accomplish the full spectrum of future operations—from high intensity, global conflict to humanitarian efforts.

Timely projection of NATO forces is required for mutual support or humanitarian aid. This is achieved through rapid mobility and increases in operational tempo. Advance notice may not be available. Changes to tasking may occur while forces are deploying. Concepts such as time-critical targeting and subsequent planning/re-planning of the mission by integrating C4I as a part of the weapons system must be supported. A complex combat mission scenario is depicted in Figure 1. Seamless integration of military capabilities will be needed to retain effectiveness.

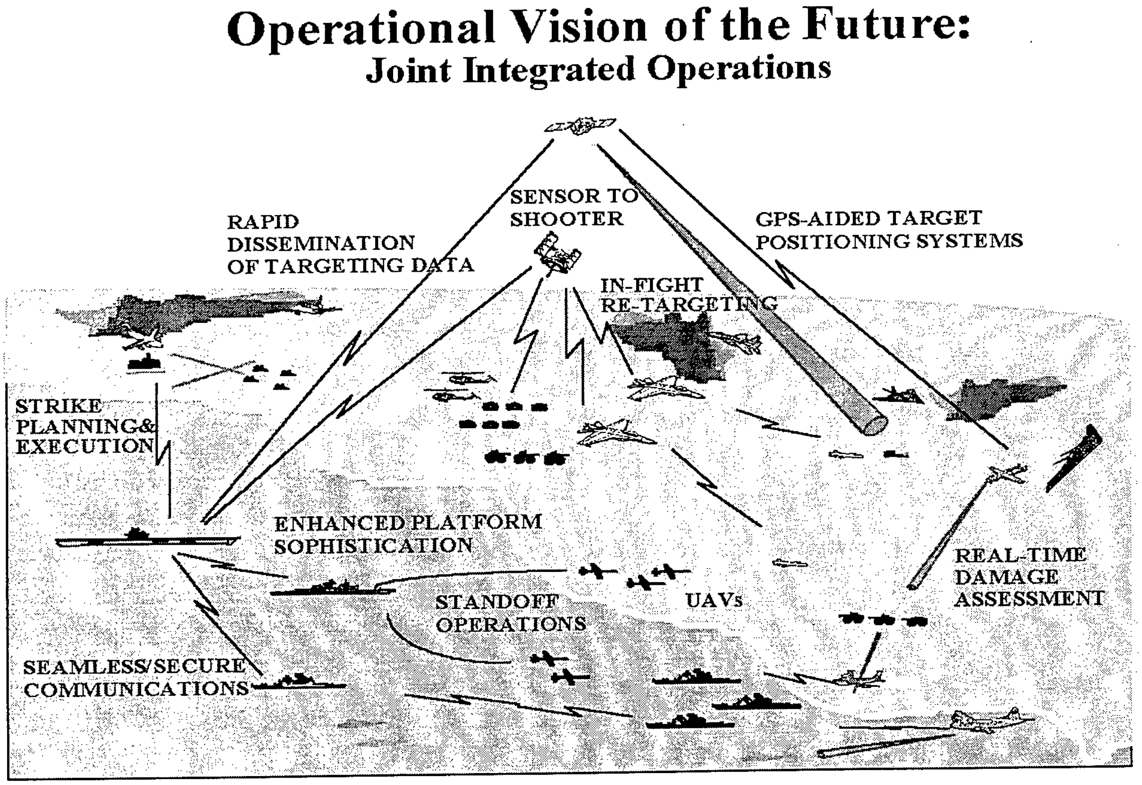

Operational Vision of the Future:
Joint Integrated Operations
RAPID DISSEMINATION OF TARGETING DATA
SENSOR TO SHOOTER
GPS-AIDED TARGET POSITIONING SYSTEMS
IN-FIGHT RE-TARGETING
STRIKE PLANNING& EXECUTION
ENHANCED PLATFORM SOPHISTICATION
STANDOFF OPERATIONS
UAVs
REAL-TIME DAMAGE ASSESSMENT
SEAMLESS/SECURE COMMUNICATIONS

Figure 1. Operational Vision of the Future

Figure 2 gives a simplified depiction of the concept. Allied warfighters are observing a common view of the battlespace (CVBS). Some may be located in the same facility while others may be viewing the CVBS on an identical display at other locations. Some may even be combatants involved in the execution of the mission, e.g., in the cockpit of an aircraft. Linked by a global communications grid, they are able to freely exchange information and collaborative plans and actions. Likewise, processed sensor information is continually fed through the grid to update the CVBS. Other users and suppliers of information also share the CVBS.

Elements of the C2 Operational Vision

From an operational viewpoint, there are three primary elements of an integrated C2 system as shown in Figure 3:

1. Global Communications Grid
2. Tailored Situational Awareness
3. Dynamic Planning and Execution

Global Communications Grid

The global communications grid functions as an information utility. It consists of interconnected networks operating as one global, high availability, secure network. It incorporates mixed media, e.g. wireline, wireless telephony, tactical radios, and satellite communications. Users can be fixed, mobile, or deployed.

The Alliance communications grid would provide peacetime, crisis, and conflict support for combat forces, logistics, and operations other than war. The grid must provide for robust net management with capability to automatically configure in some situations. Also, the networks must be fault tolerant, robust, and protected.

The rapid growth of global communications within the commercial sector provides a strong basis for such a grid. Military operations require the addition of such features as information securiy, precedence, and graceful degradation in the face of attack

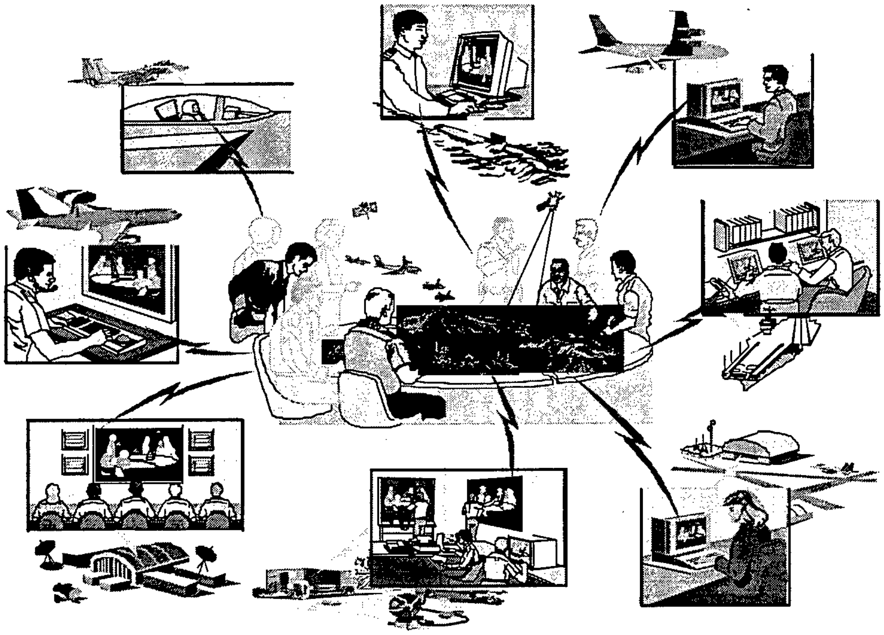


Figure 2. C2 Operational Vision of the Future Joint Integrated Operations

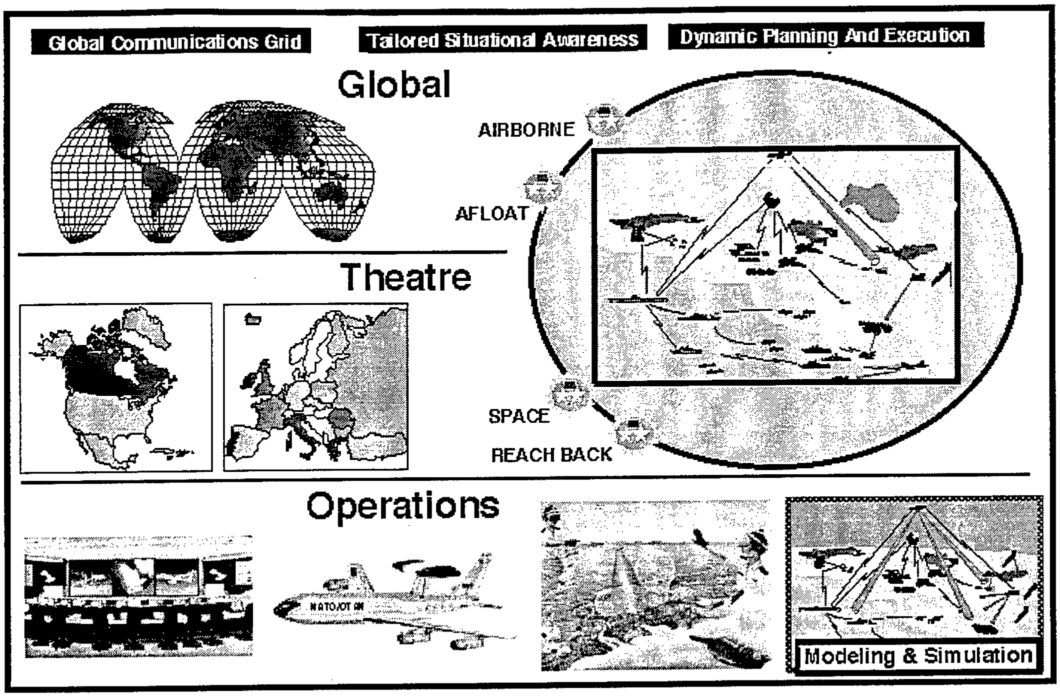

Global Communications Grid
Tailored Situational Awareness
Dynamic Planning And Execution
Global
AIRBORNE
AFLOAT
Theatre
SPACE
REACH BACK
Operations
NATO/OTAN
Modeling & Simulation

Figure 3. Primary Elements of Integrated C2

Tailored Situational Awareness
Tailored Situational Awareness provides a common view of the battlespace (CVBS) for users ranging from the highest levels of command to individual users. The CVBS is constructed from collecting data from all available sensor platforms, (i.e., space, air, ground, surface, subsurface). Then correlation and fusion of the sensor inputs creates the CVBS. Various elements of the CVBS would be "pushed" to or "pulled" by different users, and the presentation of the information is tailored to the individual user's needs and preferences. Combined force commanders, strike cell planners, tanker and AWACS pilots, strike package pilots, for example, would all be interested in enemy SAM information. Their interests would require different presentation of the information at different times. Order of Battle, asset status, plans, and execution status are all available in the CVBS.

Dynamic Planning and Execution
The planning and execution of military operations must allow for geographically separated commanders to collaborate at levels ranging from campaign to detailed missions. The planning and execution elements of the integrated C2 system provide tools to carry out offensive and defensive planning and replanning, weapons control, and combat support. These tools include common packages such as decision aids, schedulers, and targeting aids.

In summary, the C2 operational vision is realized by three complementary elements acting to provide a collaborative and distributed environment for commanders to carry out military missions. All users are provided tailored situation awareness and dynamic planning and execution functionality enabled by a global communications grid.

C2 TECHNICAL VISION
The size and complexity of the integrated C2 system that we have described herein, is unprecedented. This problem is large when we consider the scope to include multiple units and levels of command within NATO and coalition partners. Because of this complexity, system engineering, software engineering, and systems integration methodologies to address the problem are equally unprecedented.

We desire that the integrated C2 system be configurable to a number of different contingencies. The resultant composition must be able to support operations that are not only fixed, but deployable. Additionally, the resultant system should be configurable to large as well as small operations. Thus, our goal is to allow for the composition of components that fulfill the needs of multiple users in a flexible and timely manner.

The process is further complicated in that the integrated C2 system is, today, a set of existing, legacy systems. We cannot simply replace all the systems at once; we need to devise a transition that accounts for changing technology, changes in CONOPS, and the thoughtful decomposition and analysis of today's capabilities into a set of "building blocks" (components) that will enable the construct of a flexible and tailorable system.

The question becomes " How do we divide the problem into small enough pieces with well defined interfaces so as to build components and then assemble these components into various systems?" An implication of this question is "do today's methodologies scale to the dimensions (complexity) of the system that we propose to build?", and will the technologies that are available today be suitable to enable the evolutionary construction of these complex systems?

Architecture is a key item in this process. A well-defined architecture must address multiple views of this domain to include operational, system, as well as technical. Engineering large-scale systems is different from programming in the small. Instead, components must become software building blocks. Component based development of software has become an area of intense research and commercial focus resulting in several component interoperability models, such as CORBA, Active X, and JavaBeans. Additionally, enabling new technologies such as UML, Java, 4Gls, web technology, Frameworks (commercial: Netscape and Microsoft Foundation Class, the DoDs DII/COE) hold promise as technical enablers that will allow for the construct of such unprecedented C2 systems.

In October 1995, the U.S. Deputy Secretary of Defense directed that a DOD-wide effort be undertaken "to define and develop better means and processes for ensuring that C4I capabilities meet the needs of WarFighters." To accomplish that goal, *C4ISR Architecture Framework* was developed to provide a basis from which the community could work collectively to evolve and mature architecture development concepts and promulgate them as DOD direction.

Central to the C4ISR Architecture Framework was the establishment of three major perspectives, i.e., views that logically combine to describe an architecture. Those fundamental architectural descriptions were termed Operational, Systems, and Technical Architectures. C2 community feedback to that approach included the comment that, "additional products are needed to describe the Systems Architecture view." In response to that comment, a C4ISR Architecture Working Group formulated a set of essential and supporting "products," e.g., Operational Information Exchange Matrix, Logical Data Models, and a System Evolution Description.

The C4ISR Architecture Framework (See Figure 4) provides direction on how to *describe* architectures. It does not provide guidance on how to design nor implement a specific architecture or how to develop and acquire a system of systems.

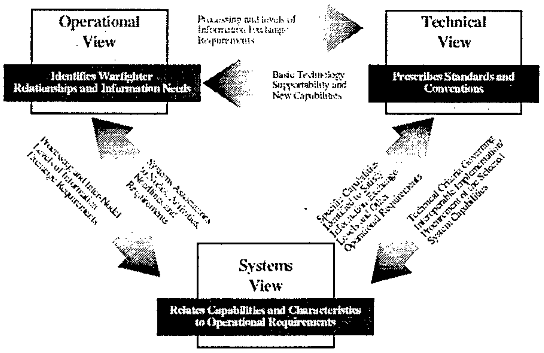

Operational View
Identifies Warfighter Relationships and Information Needs
Processing and levels of Information Exchange Requirements
Basic Technology Supportability and New Capabilities
Technical View
Prescribes Standards and Conventions
Processing and Inter-Nodal Levels of Information Exchange Requirements
Systems Associations to Nodes, Activities, Needlines and Requirements
Specific Capabilities Identified to Satisfy Information-Exchange Levels and Other Operational Requirements
Technical Criteria Governing Interoperable Implementation/ Procurement of the Selected System Capabilities
Systems View
Relates Capabilities and Characteristics to Operational Requirements

Figure 4. C4ISR Framework

One instantiation of the C4ISR Architecture Framework is the C2 System Target Architecture (C2STA) under development at Air Force Material Command Electronic System Center. The C2STA provides guidance to the acquisition community to achieve the objectives of enhancing interoperability and facilitating the fielding of flexible, reusable system components and capabilities. As such, the C2STA is the basis for moving present-day systems, and on-going and future programs, toward the C2 technical vision as shown in Figure 5.

System Composition

Architectures enable developers to concentrate on the big picture in developing a system and to adopt a component-based development philosophy as opposed to always building the system from scratch. Architectures do this by making the software systems structure explicit, providing a high level representation of a system that can be analyzed and changed before any changes are effected in implementation.

Building blocks of architectures can be thought of as components (computational elements) and connectors (interconnection and communication elements). Separating components and connectors allows for constructing flexible and scaleable systems that can evolve before and during runtime.

Existing component middleware technologies such as CORBA and Active X are component-centric; that is, they are concerned with the standardization of external properties (interfaces, bindings, and communication protocols). Software architectures, on the other hand, are system-centric; they focus on the specification of sub-systems as black-box components. One must analyze the system properties and glue code to bind system components together.

To achieve the construction of such unprecedented C2 systems that we speak, it is essential that a technology be employed that depicts the high level system representations while capturing the recurring properties of the application domain. These two aspects (system and component) and the technical disciplines that drive each, do not have a history of being adequately coupled and must be overcome if we are to be successful.

Application Frameworks have been around for some time and have been associated with the concept of component building blocks. Examples of these frameworks in the commercial world include Netscape Plugins, CORBA, DCOM, and COM+. The U.S. Defense Information Infrastructure-Common Operating Environment (DII-COE) shown in Figure 6 is an example from the military world discussed further on in this section. One discovers that when Object Oriented technology is applied to frameworks it becomes an important reuse technology and an important technology that may be useful in helping to unify system architecture and component architecture design and construction. Therefore, in the context of large-scale system development, we should look at frameworks as an important piece of the reusable design process.

Frameworks are a component in the sense that vendors sell them as products. A complex C2 application may use several frameworks in its construction. Frameworks are more customizable than most components but have more complex interfaces. Because of this complexity, the use of frameworks will require more training for designers and developers. However, frameworks are powerful and can be used by numerous applications (high reuse), and additionally, can reduce the amount of effort to develop complex applications.

Fortunately, the next generation of Object Oriented application frameworks are targeting complex business and application domains. At the heart of this effort are Object Request Broker (ORB) frameworks, which facilitate communication between local and remote objects. ORB frameworks eliminate many tedious, error-prone, and non-portable aspects of creating and managing distributed applications and reusable service components. This enables programmers to develop and deploy complex applications rapidly and robustly, rather than wrestling endlessly with low-level infrastructure concerns.

The following is a useful way to consider frameworks:

> ***System infrastructure frameworks*** simplify the development of portable and efficient system items, such as operating systems, communication frameworks, and frameworks for user interfaces. System infrastructure frameworks are primarily used internally within a software organization and are not sold to customers directly.
>
> ***Middleware integration frameworks*** are commonly used to integrate distributed applications and components. Middleware integration frameworks are designed to enhance the ability of software developers to modularize, reuse, and extend their software infrastructure to work seamlessly in a distributed environment. Middleware integration frameworks represent a thriving market, and are rapidly becoming commodities. Common examples include CORBA-based middleware, message-oriented middleware, and distributed transactional middleware.

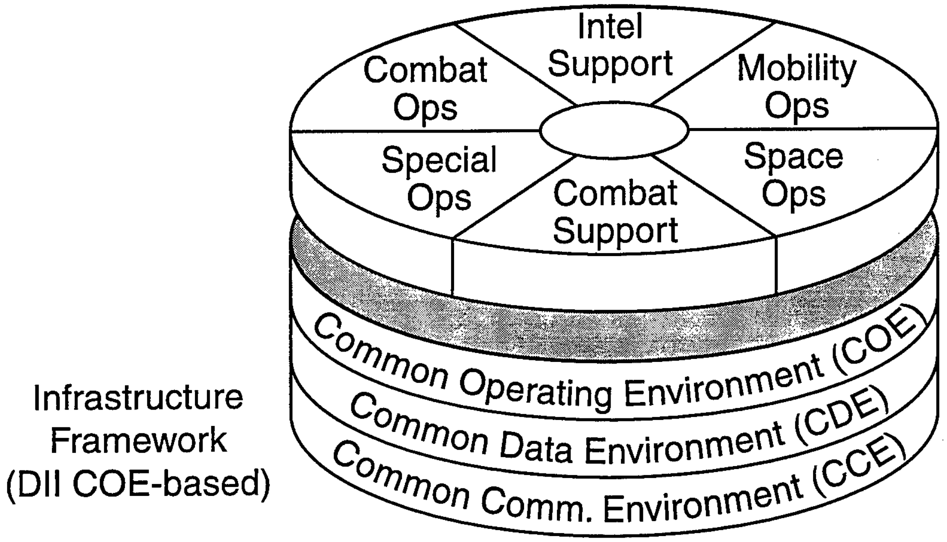

Intel Support
Combat Ops
Mobility Ops
Special Ops
Combat Support
Space Ops
Infrastructure Framework (DII COE-based)
Common Operating Environment (COE)
Common Data Environment (CDE)
Common Comm. Environment (CCE)

Figure 5. C2 System Target Architecture

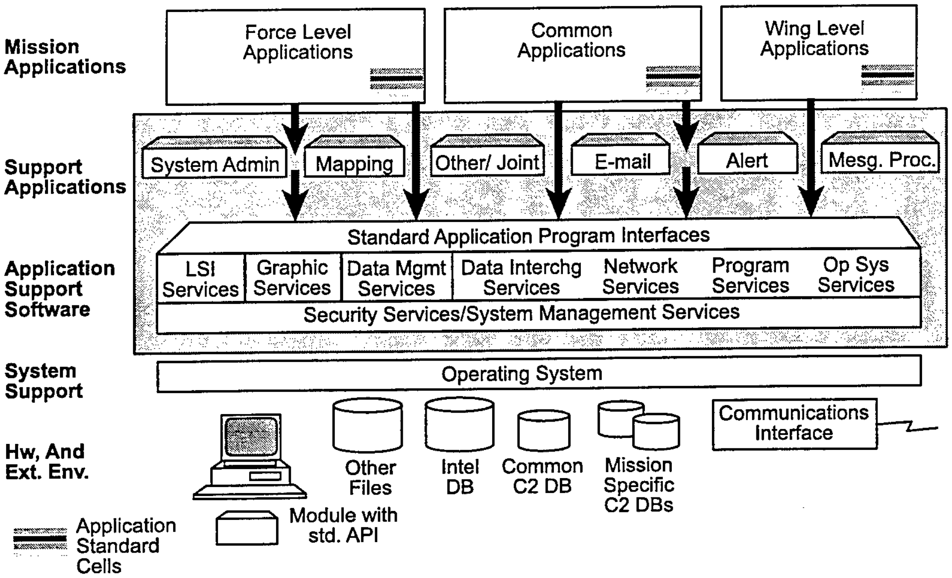


Figure 6. Common Operating Environment

Enterprise application frameworks address broad application domains (such as telecommunications, avionics, manufacturing, and financial engineering and are the cornerstone of enterprise business activities. Relative to system infrastructure and middleware integration frameworks, enterprise frameworks are expensive to develop and/or purchase. However, enterprise frameworks can provide a substantial return of investment since they support the development of end-user applications and products directly.

Some Enabling Technologies

Operating Environments

Using the definitions above we can consider the Defense Information Infrastructure-Common Operating Environment (DII-COE) as an instantiation of a framework that serves the needs of a system infrastructure framework while exhibiting properties of a middleware integration framework as well.

The goal of the DII-COE is to provide a consistent set of tools for installation, administration and common applications (e.g., office automation, message handling) all running on a common system kernel. COTS software is the preferred COE implementation approach.

CORBA/DCOM

Many of the fundamental issues associated with the problem domain can be addressed through the use of a brokered architecture such as CORBA. Using CORBA, systems communicate by plugging into a brokered framework via a formal interface description language (IDL). IDL provides a common way of accessing not only the capabilities of the communicating systems but also the capabilities provided by the framework. This approach is particularly well suited to the integration of legacy systems, since such legacy systems can be encapsulated with a small amount of new software that provides an IDL interface.

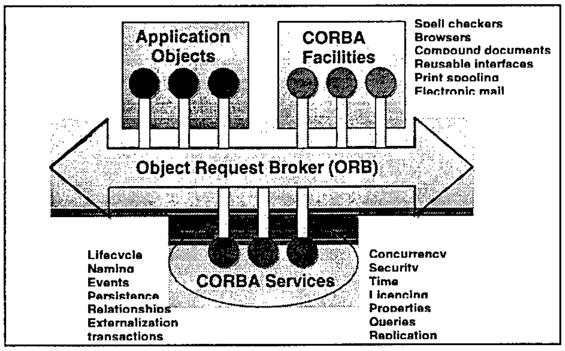


Figure 7. CORBA Reference Model

The CORBA reference model is shown Figure 7. This model also allows services common to several objects or applications to be made available as an adjunct to the basic request broker functionality (in effect extending the infrastructure). The CORBA reference model includes Object Services and Common Facilities, both of which represent generally useful functionality and which are intended to be provided by COTS vendors. This allows application developers (or integrators) to focus on providing functionality that is unique to the business domain without having to provide such common services.

Thus, CORBA provides the primary advantages of:

Hardware platform and language independence: it does not matter what language or platform is used to develop or reference (invoke) the objects

Network location transparency: requests look as though they are performed local to the requesting process.

Increased fault tolerance: the ability to transparently start or restart services on demand or to bind to a different service.

Polymorphism: the ability to invoke the same method on different objects causing different behavior.

Reduced development effort: the ability to leverage common services and facilities provided by commercial vendors.

Consistent formalism for interfaces: all objects at all levels of granularity are described in a consistent way, as opposed to the diverse interface technologies characteristic of legacy systems.

Consistent documentation: IDL interfaces can be understood by humans (perhaps with the aid of visual browsers), and at the same time can be uniform across implementations.

JAVA Technologies

Java Virtual Machines hold the promise of removing existing hardware dependencies on new Command and Control software. By building to a common virtual machine specification, only a single runtime of a C2 function would need to be built. The virtual machine allows this code to be run on a number of different hardware platforms without change.

Java Remote Method Invocation (RMI) enables programmers to create distributed Java-to-Java applications, in which the methods of remote Java objects can be invoked from other Java virtual machines, possibly on different hosts.

Web Technology

The world is experiencing a revolution in the distribution and viewing of information enabled by very rapid paced improvements in web technology. The COTS world is the driver in this area. Web technologies which bring usable timely data to C2 decision makers are relatively inexpensive "force multipliers" in the integration of Command and Control systems. Web technology forces the developer to separate the visual presentation from calculation of the data, thus improving the flexibility and portability of C2 applications. The same application can present different but consistent presentations in the native language of the user

Stovepipes to Components

Today's technical architectures reflect design approaches of the 70's and 80's. Systems built in this environment are difficult and expensive to modify and maintain. In addition, they tend not to be interoperable. Each system developed its communication, situation awareness, and planning execution software functionality independently, and redundantly. Systems with these characteristics are commonly referred to as "stovepipe" systems.

As previously mentioned, the use of a common architecture and frameworks changes the development focus to components. The next challenge we must face is changing the acquisition methods to take advantage of the aforementioned technological opportunities while transitioning existing stovepipes to components.

ACQUISITION METHODOLOGY

The first step in the methodology of system development is to consolidate, refine and optimize mission needs in terms of joint and coalition requirements into an evolutionary process that validates needs, prioritizes development efforts, allocates funding and determines field readiness. In this way, we replace the develop/refine/field/integrate approach with a develop/integrate/field/refine approach. This gets capability into the field as soon as possible. As the users gain operational capability, they become more comfortable with the system, revise their requirements and add new ones. These features are then added a few at a time and the cycle repeats.

The system gains maturity in operation. New operations concepts and system requirements are assimilated into the ongoing development process and new technologies are rapidly applied. The guiding principles of this approach are commercial in their origin. We use commercial equipment and practice to the greatest extent possible, the users are kept involved in every step of the process. We encourage innovation, competition, and rely on the marketplace to drive out the best solutions.

Spiral Development

Our goal is to get functionality to the field more quickly. This will be achieved by decreasing acquisition cycle time via an evolutionary spiral development process, as shown in Figure 8. User requirements combined with technology (COTS/GOTS) will merge to form prototypes that will be tested, demonstrated, and integrated in a Technology Insertion Facility. Concepts for the usage of these prototypes will be defined and validated in Battle Laboratories by the user community. A Technology Insertion Facility allows experimentation with new technologies using real data. The function of a Battle Laboratory is to experiment, refine and document Concepts of Operations (CONOPS) for the operational use of a previously developed capability or functionality. The Battle Laboratory does not experiment with technologies, but instead, experiments, tests, refines and documents the operational changes and improvements achieved with new technologies.

Testing in Battle Laboratories may preclude the premature fielding of prototypes to Forward Operating Locations (FOL). When experimental systems are employed at a FOL, they require continued support and training to remain "operational". This practice disrupts operations and makes some FOL's resemble collections of interoperable and stand-alone prototypes.

After each functional capability passes through the Technology Insertion Facility and Battle Labs, it is evaluated for possible fielding. If the capability is not deemed mature enough for fielding it will enter another spiral cycle were it is further refined. Functional capabilities that show no promise are eliminated early before extensive time and money is expended.

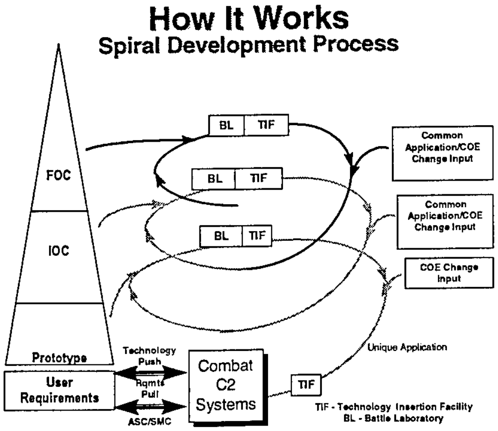


Figure 8. Spiral Development Process

Demos and Experiments Identify Spiral Candidates

New technology candidates are identified at larger organized events, such as US Department of Defense's JWID (Joint Warrior Interoperability Demonstration) or US Air Force EFX (Expeditionary Force Experiment). These candidates are demonstrated or experimented with in an effort to determine their efficacy in the Command and Control environment. The most mature candidates may prove so valuable that early deployment is forced. These and the lesser mature candidates will enter the spiral development process for further refinements. The least mature candidates can be revisited in future events.

ROUTE AHEAD - INVESTMENT STRATEGY FOR NATO

The following summarizes key aspects of a strategy that could be beneficial in achieving an integrated C2 system:

1. The basic tenets for system acquisition need to changed to accommodate the rapid change of commercial technology as well as the need to move to more integrated systems quickly. Unless our ability to acquire capabilities in a more timely and flexible manner changes, we will continue to field dated systems.

2. Investments in Software Component Technology that contribute to system composition should be initiated. To a large extent, we should analyze and capitalize on commercial technology to accomplish this.

3. Mechanisms for migration from stovepipes to components should be sought. An overall technical insertion program needs to be devised to accelerate the fielding of new capabilities utilizing elements of legacy systems.

ACKNOWLEDGMENT

The authors thank Edwin J. Greene, Ross W. Wheelwright, Robert L. Pancotti, Stephen C. Schwarm, and Kenneth Brayer of The MITRE Corporation for their assistance in preparing this paper. We would also like to thank Dr. Harold W. Sorenson of MITRE for his leadership in establishing the vision for integrated command and control.

REFERENCES

1. Woodall, D., Evolution of Mission Management Integration, AGARD Spring Symposium of Mission Systems Panel, Saclay, France, p 2-1 to 2-7, 1997

Information Processing Architecture for Mission Performance of Autonomous Systems Capable of Dynamic Vision

Ernst Dieter Dickmanns, Simon Fürst
Universitaet der Bundeswehr Munich (UBM)
Institut fuer Systemdynamik und Flugmechanik (ISF)
D-85577 Neubiberg, Germany
e-mail: Ernst.Dickmanns@unibw-muenchen.de

ABSTRACT

Dynamic machine vision under development will allow unprecedented flexibility in automation of mission performance, maybe even approaching human capabilities in the long run. Contrary to conventional measurements, vision provides information not just on a single object, but on all objects in the field of view. Saccadic vision with a foveal / peripheral layout of the complex vehicle eye combines a large field of view with a central area of high spatial resolution; this area may be shifted almost momentarily at will to points of special interest. The information processing architecture and the advantages of a system design corresponding to these guidelines are discussed. First results of a mission based on near Earth landmark navigation are given.

1. INTRODUCTION

Autonomous systems capable of dynamic vision are under development for a little more than a decade by now (Ref. 1,2). First results in real world applications have been achieved in the second half of the 80ies; relative state estimation in real landing approaches has been demonstrated in 1991 for the first time (Ref. 3). Due to missing computing power in small transportable units it will take at least another decade until a state of development will be achieved that allows interpreting multiple video data streams in real-time for robust perception also under perturbed conditions in natural environments.

The sensors will not be confined to the optical range of the human eyes. Both infra-red, low-light-level-TV and imaging radar will extend the range of operation considerably. All-weather and day/night operations will be most likely, thereby complementing human capabilities.

What will be the benefit of machine vision for mission performance? It is not felt to be the most appropriate way to further overburden the human pilot by having him interpret additional image data. Instead, it is conjectured that machine vision systems on their own should develop similar capabilities in dynamic scene understanding as those human pilots have. This requires an interpretation of image sequences both in 3-D space and over time simultaneously.

The 4-D approach to machine vision developed at UBM over the last two decades (Ref. 1-5) extends recursive estimation (derivatives of the well-known Kalman filter) to perspective image sequence interpretation. Dynamical models of the processes observed are not only used for the individual estimation processes but also for short and longer term predictions of relevant objects in conjunction (so-called situation assessment) which allows deeper understanding of what is being observed, especially, if the notion of subjects is fully exploited.

Subjects are objects capable of self-induced motion control depending on their internal state which may be influenced by their perception capabilities, most notably their own sense of vision (Ref. 6). Subjects try to achieve goals and may be characterized by their stereotypical behaviors to this end in certain situations. Knowledge about various classes of subjects, biological and technical ones alike, is essential for deeper understanding of 'the world'.

The time it will take until sufficient computing power in small transportable units will be available should be used for developing corresponding software and these knowledge bases; at UBM, the 'Expectation-based, Multi-focal Saccadic (EMS-) Vision' system under development right now is expected to serve these purposes. The next section 2 will give a survey on the efforts towards more powerful vehicle eyes at UBM; then, in section 3 the general architecture for generating behavioral capabilities from state representations of several objects will be discussed.

The resulting perceptual and behavioral capabilities define the decomposition of missions into mission elements depending on types of activities required. Sensor information available allows to recognize typical situations; these situations define control activities allowed and likely to be successful. Switching between behavioral modes requires the capability of recognizing events and of keeping track of changing situations; this means that the larger context between different objects, own goals and their temporal evolution has to be evaluated according to relevant background knowledge.
Section 4 is devoted to situation assessment centered about the individual subject; section 5 will then reassess the notion of situations as seen from a level higher up in the task hierarchy. In section 6, aspects of system integration and first results achieved in mission performance with hardware-in-the-loop simulations will be discussed.

2. VISUAL PERCEPTION OF SITUATIONS

In order to be flexible and adaptable to changing contexts, it is not sufficient to track the one object of highest interest at a time; instead, superiority in handling complex situations requires that a large field of view and high resolution at a central part are available simultaneously. The large field of view allows rough tracking of several objects of relevance in a certain situation; the region of high resolution, the targeting of which has to be controlled by the interpretation process, should allow sufficient details on the one object of highest interest.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

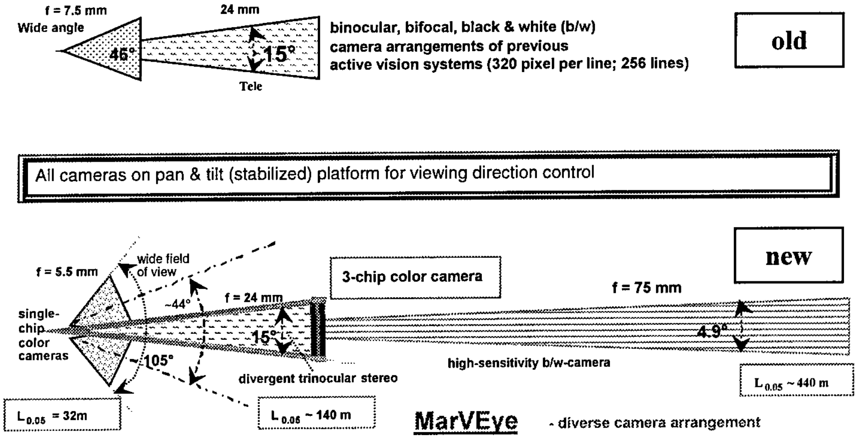

f = 7.5 mm
Wide angle
45°
24 mm
15°
Tele
binocular, bifocal, black & white (b/w)
camera arrangements of previous
active vision systems (320 pixel per line; 256 lines)
old
All cameras on pan & tilt (stabilized) platform for viewing direction control
f = 5.5 mm
wide field of view
~44°
f = 24 mm
single-chip color cameras
15°
~105°
divergent trinocular stereo
3-chip color camera
f = 75 mm
4.9°
high-sensitivity b/w-camera
new
$L_{0.05}$ = 32m
$L_{0.05}$ ~ 140 m
$L_{0.05}$ ~ 440 m
MarVEye
- diverse camera arrangement

Figure 1: Arrangement of four CCD-cameras for a complex vehicle-eye yielding a large simultaneous field of view (peripheral vision), a high-resolution central area (foveal vision), and trinocular stereo capability for the near range.

Figure 1 shows the camera arrangement 'MarVEye' as proposed for road vehicle applications (Ref. 7). The focal lengths and the camera sensitivities used may be adapted to the mission at hand; the entire unit can be viewing direction controlled from the image evaluation process. This system is meant to be a (poor) technical equivalent to the vertebrate eye in biology; there are several control modes available like search, inertial stabilization, smooth pursuit (visual fixation) and saccades.

2.1 Visual perception of single objects

Since the late 70ies, observer techniques as developed in systems dynamics (Ref. 8) have been used at UBM in the field of motion control by computer vision (Ref. 9). In the early 80ies, H.J. Wuensche did a thorough comparison between observer- and Kalman filter realizations in recursive estimation applied to vision for the original task of balancing an inverted pendulum on an electro-cart by computer vision (Ref. 10). Since then, refined versions of the Extended Kalman Filter (EKF) with numerical stabilization (UDU^T-factorization, square root formulation) and sequential updates after each new measurement have been applied as standard methods to all dynamic vision problems at UBM.

Based on experience gained from 'satellite docking', road vehicle guidance, and on-board autonomous aircraft landing approaches by machine vision, it was realized in the mid 80ies, that the joint use of dynamical models and temporal predictions for several aspects of the overall problem in parallel was key to achieving a quantum jump in the performance level of autonomous systems based on machine vision. Beside state estimation for the physical objects observed and control computation based on these estimated states, it was the feedback of knowledge thus gained to image feature extraction and to the feature aggregation level which allowed for an increase in efficiency of image sequence evaluation of one to two orders of magnitude. (See fig.2 for a graphical overview.)

Following state prediction, the shape and the measurement models were exploited for determining:

- viewing direction control by pointing the two-axis platform carrying the cameras;
- locations in the image where information for most easy, non-ambiguous and accurate state estimation could be found (feature selection),
- the orientation of edge features which allowed to reduce the number of search masks and directions for robust yet efficient and precise edge localization,
- the length of the search path as function of the actual measurement uncertainty,
- strategies for efficient feature aggregation guided by the idea of the 'Gestalt' of objects, and
- elements in the Jacobian matrices (first order derivatives of feature positions relative to state components in the dynamical models) which contain rich information for interpretation of the motion process in a least squares error sense, given the motion constraints, the features measured, and the statistical properties known.

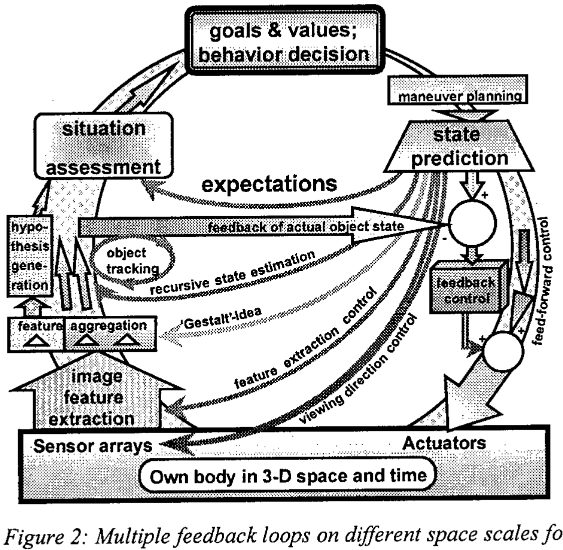


Figure 2: Multiple feedback loops on different space scales for efficient scene interpretation and behavior control: control of image acquisition and -processing (lower left corner), 3-D 'imagination'-space in upper half; motion control (lower right corner).

This integral use of

1. dynamical models for motion of and around the center of gravity taking actual control outputs and time delays into account,
2. spatial (3-D) shape models for specifying visually measurable features,
3. the perspective mapping models, and
4. prediction error feedback for estimation of the object state in 3-D space and time

simultaneously and in closed loop form was termed the '4-D approach'. It is far more than a recursive estimation algorithm based on some arbitrary model assumption in some arbitrary subspace or in the image plane.

Initially, in our applications just the ego-vehicle has been assumed to be moving on a smooth surface or trajectory, with the cameras fixed to the vehicle body. In the meantime, solutions to rather general scenarios are available with several cameras spatially arranged on a platform, which may be pointed by voluntary control relative to the vehicle body. These camera arrangements allow a wide simultaneous field of view, a central area for trinocular (skew) stereo interpretation, and a small area with high image resolution for 'tele'-vision. The vehicle may move in full 6 degrees of freedom; while moving, several other objects may move independently in front of a stationary background. One of these objects may be 'fixated' (tracked) by the pointing device using inertial and visual feedback signals for keeping the object (almost) centered in the high-resolution image. An object newly appearing in the wide field of view may trigger a fast viewing direction change such that this object can be analyzed in more detail by one of the tele-cameras. This corresponds to 'saccadic' vision as known from vertebrates and allows very much reduced data rates for a complex sense of vision. It essentially trades the need for 1. time-sliced control of attention, and 2. scene reconstruction from sampled data, for a reduction in data rate of 1 to 2 orders of magnitude (as compared to full resolution in the entire simultaneous field of view).

The 4-D approach lends itself for this type of vision since both object-orientation and the temporal ('dynamical') models are available in the system already. In the next subsection the basic assumptions underlying the 4-D approach are summarized.

2.1.1 *Basic assumptions underlying the 4-D approach*

It is the explicit goal of this approach to take, as much as possible, advantage of physical and mathematical models of processes happening in the real world. Models developed in the natural sciences and in engineering over the last centuries, in simulation technology and in systems engineering (decision and control) over the last decades form the base for computer-internal representations of real-world processes:

1. The (mesoscopic) world observed happens in **3-D space and time** as the independent variables; non-relativistic (Newtonian) models are sufficient for describing these processes.

2. All interactions with the real world happen ***'here and now'***, at the location of the body carrying special input/output devices. Especially, the locations of the sensors (for signal or data input) and of the actuators (for control output) as well as those body regions with strongest interaction with the world (as for example the wheels of ground vehicles) are of highest importance.

3. **Efficient interpretation of sensor signals** requires background knowledge about the (motion) ***processes*** observed and controlled, that is both its spatial and temporal characteristics. Invariants for process understanding may be abstract model components not graspable at one point in time.

4. Similarly, **efficient computation of** (favorable or optimal) **control outputs** can only be done taking complete (or partial) process models into account; control theory provides the methods for fast and stable reactions.

5. **Wise behavioral decisions** require knowledge about the longer-term outcome of special feed-forward or feedback control modes in certain situations and environments; these results are obtained from integration of the dynamical models. This may have been done beforehand and stored appropriately, or may be done on the spot if analytical solutions are available or numerical ones can be derived in a small fraction of real-time as becomes possible now with the increasing processing power available. Single behaviors are realized by triggering the behavioral modes discussed under point 4 above.

6. **Situations** are made up of arrangements of objects, other active subjects, and of the own goals pursued (see section 4 and 5 below); therefore,

7. it is essential to recognize **single objects and subjects**, their relative state, and for the latter also, if possible, their intentions in order to be able to make meaningful predictions about the future development of a situation (which is needed for successful behavioral decisions).

8. As the term ***re*-cognition** tells, in the usual case it is assumed that objects seen are (at least) generically known already. Only their appearance here (in the geometrical range of operation of the senses) and now is new; this allows a fast jump to an object hypothesis when first visual impressions arrive through sets of features. Exploiting background knowledge, the model based perception process has to be initiated. Free parameters in the generic object models may be determined and efficiently adjusted dynamically by attention control and the use of special algorithms and behaviors.

9. In order to be able to do step 8 efficiently. knowledge about 'the world' has to be provided in the **context of 'task domains'** in which likely co-occurrences are represented. In addition, knowledge about discriminating features is essential for correct hypothesis generation (indexing into the object database).

10. Most efficient descriptions of objects (or object classes) by **invariants** are usually done **in 3-D space** (for shape) **and time** (for motion constraints or stereotypical motion sequences). Modern microprocessors are sufficiently powerful to compute the visual appearance of an object under given aspect conditions in an image (in a single one, or even in several ones with different mapping parameters in parallel) at runtime. They are even powerful enough to numerically compute the Jacobian matrices for sensor/object pairs of features evaluated with respect to object state or parameter values; this allows a very flexible general framework for recursive state and parameter estimation. The inversion of perspective projection is thus reduced to a least squares model fit once the recursive process has been started. The underlying assumption here is that local linearizations of the overall process are sufficiently good representations of the nonlinear real process; for high

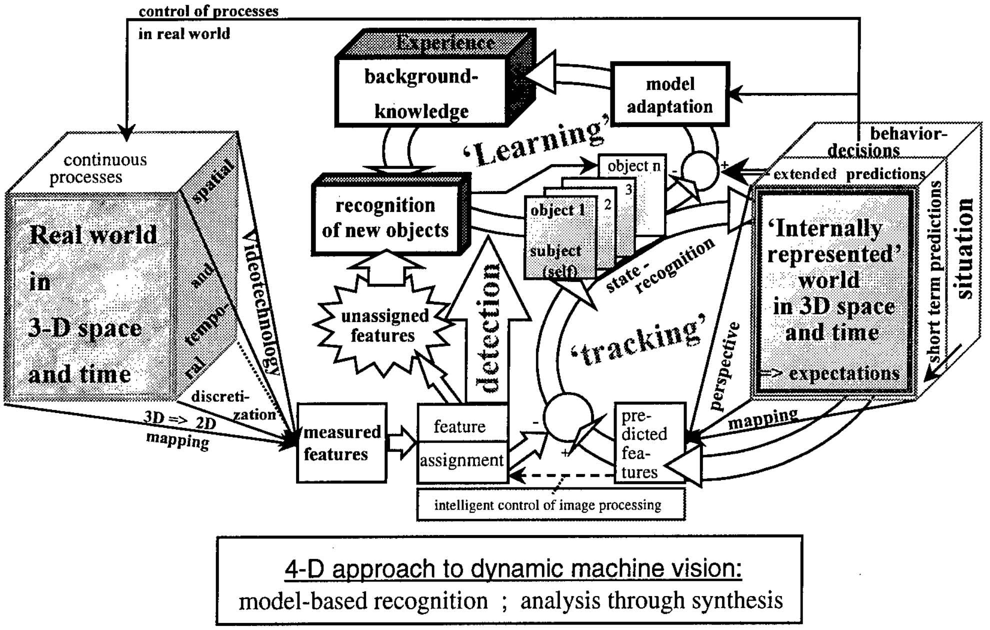


Figure 3: Survey on the 4-D approach to dynamic machine vision with three major areas of activity: Object detection (central arrow upwards), tracking and state estimation (recursive loop in lower right), and learning (loop in center top), the latter two being driven by prediction error feedback.

evaluation rates like video frequency (25 or 30 Hz) this is usually the case.

11. In a running interpretation process of a dynamic scene, **newly appearing objects** will occur **in restricted areas** of the image such that bottom-up search processes may be confined to these areas. Passing cars, for example, always enter the field of view from the side just above the ground; a small class of features allows detecting them reliably.

12. **Subjects**, i.e. objects with the capability of self-induced generation of control actuation, are characterized by typical (sometimes stereotypical, i.e. predictive) motion behavior in certain situations. This may also be used for recognizing them (similar to shape in the spatial domain).

13. The same object/subject may be represented internally at **different scales with various degrees of detail**; this allows flexible and efficient use in changing contexts (e.g. as a function of distance or degree of attention).

2.1.2 *Basic idea of the 4-D approach to dynamic vision*
Figure 3 shows the main three activities running in parallel in an advanced version of the 4-D approach:

1. **Detection of objects** from typical collections of features not yet assigned to some object already tracked (center left, upward arrow). When these feature collections are stable over several frames, an object hypothesis has to be formed, and the new object is added to the list of those regularly tracked (arrow to the right).

2. **Tracking of objects and state estimation** is shown in the loop to the lower right in figure 3; first, with the control output chosen, a single step prediction is done in 3-D space and time, the 'imagined real world'. This step consists of two components, a) the 'where'- signal path concentrating on progress of motion in both translational and rotational degrees of freedom, and b) the 'what'- signal path dealing with object shape. (In order not to overburden the figure these components are not shown.)

3. **Learning from observation** is done with the same data as used in tracking; however, this is not a single step loop but rather a low frequency estimation component concentrating on 'constant' parameters, or it even is an off-line component with batch processing of stored data. This is an actual construction site in code development, at present, which will open up the architecture towards becoming more autonomous in new task domains as experience of the system grows. Both dynamical models (for the 'where'-part) and shape models (for the 'what'-part shall be learnable.
Another component under development not detailed in figure 3 is situation assessment and behavior decision; this will be discussed in sections 4 and 5.

2.1.3 *Generic 4-D object classes*
The efficiency of the 4-D approach to dynamic vision is achieved by associating background knowledge about classes of objects and their behavioral capabilities with the data input. This knowledge is available in generic form, that is, structural information typical for object classes is fixed while specific parameters in the models have to be adapted to the special case at hand. Motion descriptions for the center of gravity (the translational object trajectory in space) and for rotational movements, both of which together form the so-called 'where'-problem are separated from shape descriptions, called the 'what'-problem. Typically, summing and averaging of feature positions is needed to solve the 'where'-problem, while differencing feature positions contributes to solving the 'what'-problem.

2.1.3.1 *Motion description*
Possibilities for object trajectories are so abundant that they cannot be represented with reasonable effort. However, good models are usually available describing their evolution over

time as a function of the actual state, the control- and the perturbation inputs. These so-called 'dynamical models', usually, are sets of nonlinear differential equations ($\underline{x} = \underline{f}(\underline{x}, \underline{u}, \underline{v}', t)$) with $\underline{x}$ as the n-component state vector, $\underline{u}$ as r-component control vector and $\underline{v}'$ as perturbation input.

Through linearization around a nominal trajectory $\underline{x}_N(t)$, locally linearized descriptions are obtained which can be integrated analytically to yield the (approximate) local transition matrix description for small cycle times T

$$\underline{x}[(k+1)T] = A\,\underline{x}[kT] + B\,\underline{u}[kT] + \underline{v}[kT]. \qquad (1)$$

The elements of the matrices A and B are obtained from $F(t) = \partial\underline{f}/\partial\underline{x}|_N$ and $G(t) = \partial\underline{f}/\partial\underline{u}|_N$ by standard methods from systems theory.

Usually, the states cannot be measured directly but through the output variables $\underline{y}$ given by

$$\underline{y}[kT] = h(\,\underline{x}[kT], \underline{p}, kT) + \underline{w}[kT], \qquad (2)$$

where h may be a nonlinear mapping (see below), $\underline{p}$ are mapping parameters and $\underline{w}$ represents measurement noise.

On the basis of eq. (1) a distinction between 'objects' proper and 'subjects' can be made: If there is no dependence on controls $\underline{u}$ in the model, or if this $\underline{u}(t)$ is input by another agent one speaks of an 'object', controlled by a subject in the latter case. If $\underline{u}[kT]$ may be activated by some internal activity within the object, be it by pre-programmed outputs or by results obtained from processing of measurement data, one speaks of a 'subject'.

2.1.3.2 *Shape and feature description*

With respect to shape, objects and subjects are treated in the same fashion. Only rigid objects and objects consisting of several rigid parts linked by joints have been treated. Since objects may be seen at different ranges the appearance in the image may vary considerably in size. At large ranges the 3-D shape of the object, usually, is of no importance to the observer, and the cross-section seen contains most of the information for tracking. However, this cross-section depends on the angular aspect conditions; therefore, both coarse-to-fine and aspect-dependent modeling of shape is necessary for efficient dynamic vision. This will not be discussed here (see Ref. 2).

Experience tells that area based features should play an important role in robust detection and object tracking. Initially, this has been realized by observing the average gray value on the vehicle-side of edge features detected; with more computing power available, color profiles in certain cross-sections yield improved performance.

2.1.4 *Image feature extraction*

Due to space restrictions, this topic will not be detailed here; the interested reader is referred to (Ref. 1). Two types of feature extraction algorithms are used: Oriented edge features extracted by ternary mask correlations in horizontal or vertical search paths (a rather old component), and area-based segmentations of 'stripes' of certain widths, arbitrarily oriented in the image plane (a new one).

The intelligent control of the parameters of these algorithms is essential for efficient tracking. In the 4-D approach, these parameters are set by predictions from the spatio-temporal representations and application of perspective mapping. A small percentage of image data properly analyzed allows to track objects reliably and precisely when used in a tight bottom-up and top-down loop traversed frequently (25 Hz); this has to be seen in the context of figure 2.

2.1.5 *State estimation*

The basic approach has been described many times (see Ref. 1, 3, 4, 5, 10) and has remained the same for visual relative state estimation over years by now. However, in order to be able to better deal with the general case of scene recognition under (more strongly) perturbed ego-motion, an inertially based component has been added (Ref. 11, 12).

This type of state estimation is not new at all if compared to inertial navigation, e.g. for missiles; however, here only very inexpensive accelerometers and angular rate sensors are being used. This is acceptable only because the resulting drift problems are handled by a visual state estimation loop running in parallel, thereby resembling the combined use of (relatively poor) inertial signals from the vestibular apparatus and of visual signals in vertebrate perception. Some of these inertial signals may also be used for stabilizing the viewing direction with respect to the stationary environment by direct negative feedback of angular rates to the pointing device carrying the cameras. This feedback actually runs at very high rates in our systems (500 Hz, see Ref.13)

2.1.6 *Inertially based ego-state estimation with visual stabilization (IbSE)*

The advantage of this new component is threefold: 1. Because of the direct encoding of accelerations along, and rotational speed components around body fixed axes, time delays may be neglected. These components can be integrated numerically to yield predictions of positions. 2. The quantities measured correspond to the forces and moments actually exerted on the vehicle including the effects of perturbations; therefore, they are more valuable than predictions from a theoretical model disregarding perturbations which are unknown, in general. 3. If good models for the eigen-behavior are available, the inertial measurements allow the estimation of parameters in perturbation models; this may lead to deeper understanding of environmental effects.

2.1.7 *Dynamic vision*

With respect to ego-state recognition, vision now has reduced but still essential functionality. It has to stabilize long-term interpretation relative to the stationary environment, and it has to yield information on the environment, like position and orientation relative to the road and road curvature in vehicle guidance, or relative to landmarks not measurable inertially. With respect to other vehicles or obstacles, the vision task also is slightly alleviated since the high-frequency viewing direction component is known now; this reduces search range required for feature extraction and leads to higher efficiency of the overall system.

These effects can only be achieved using spatio-temporal models and perspective mapping, since these items link inertial measurements to features in the image plane. With different measurement models for all the cameras used, a single object model and its recursive iteration loop may be fed with image data from all cameras relevant. Jacobian matrices now exist for each object/sensor pair.

The nonlinear measurement equation (2) is linearized around the predicted nominal state $\underline{x}_N$ and the nominal parameter set $\underline{p}_N$ yielding (without the noise term)

$$\begin{aligned}\underline{y}[kT] &= \underline{y}_N[kT] + \underline{\delta y}[kT] \qquad (3)\\ &= \underline{h}(\,\underline{x}_N[kT], \underline{p}_N, kT) + C_x\,\underline{\delta x} + C_p\,\underline{\delta p},\end{aligned}$$

where $C_x = \partial\underline{h}/\partial\underline{x}|_N$ and $C_p = \partial\underline{h}/\partial\underline{p}|_N$ are the Jacobian matrices with respect to the state components and the parameters involved. Since the first terms to the right hand side of the equality sign are equal by definition, eq. (3) may be used to determine $\underline{\delta x}$ and $\underline{\delta p}$ in a least squares sense from $\underline{\delta y}$ as the prediction error measured (observability given); this is the core of recursive estimation.

2.2 Situations as relative states of objects in a task context

For each object an estimation loop is set up yielding best estimates for the relative state to the ego-vehicle including all spatial velocity components. For stationary landmarks, the velocity is the negative of ego-speed, of course. Since the own velocity is known reliably from conventional measurements, the distance to the landmark can be determined even with monocular vision exploiting motion stereo (Ref. 14).

With all this information available for the surrounding environment and the most essential objects in it, an interpretation process can evaluate the situation in a task context and come up with a conclusion whether to proceed with the behavioral mode running or to switch to a different mode. Fast in-advance simulations exploiting dynamical models and alternative stereotypical control inputs yield possible alternatives for the near-term evolution of the situation. By comparing the options or by resorting to pre-computed and stored results, these decisions are made.

The relative state of all objects of interest is kept in a data structure called scene tree, representing the variables in sets of homogeneous transformation matrices as known from computer graphics. From this 'dynamic object data base' all information of interest can be derived by well-known data manipulations.

3. BEHAVIORAL CAPABILITIES DEFINE MISSION DECOMPOSITION

3.1 Generation of behavioral capabilities

Dynamic vision is geared to closed-loop behavior in a task context; the types of behavior of relevance, of course, depend on the special task domain. The general aspect is that behaviors are generated by control output. There are two basically different types of control generation:

1. Triggering the activation of (generically) stored time histories, so-called feed-forward control, by events actually observed,
2. gearing actual control to the difference between desired and actual state of relevant systems, so-called feedback control.

In both cases, actual control parameters may depend on the situation given. A very general method is to combine the two given above (as a third case in the list), which is especially easy in the 4-D approach where dynamical models are already available for the part of motion understanding.

The general feed-forward control law in generic form is

$$\underline{u}(\tau) = \underline{g}(\underline{p}_M, \tau_M), \quad \text{with } 0 < \tau = t - t_{Trig} < (\tau_M); \qquad (4)$$

$\underline{p}_M$ may contain averaged state components (like speed).

A typical feed-forward control element is the lateral stick deflection angle (aileron position) control time history for a heading change with an airplane. In a generic formulation, for example, the control input is specified through three phases with a few constant parameters in each one. All three together yield the maneuver time τ_M. The first phase of duration τ_I essentially consists of an aileron pulse deflection, say a term $A[1-\cos(\omega\tau)]$ with $\omega\tau_I=2\pi$; amplitude A and frequency ω (or duration τ_I) are adaptable maneuver parameters. In the second phase of different duration τ_B, control deflection is zero, i.e. stick position δ is constant (essentially at a finite bank angle Φ and heading change rate). In the third phase, the amplitude of the control input is of opposite sign to the first one. The other parameters are essentially the same, yielding bank angle zero at the end. These parameters A, $\omega\tau_I$, τ_M, and τ_B have to be selected such that at $(\tau_M + \Delta\tau_D)$ the heading direction of the aircraft is equal to the desired one; the magnitude of the parameters, of course, depends on the speed flown and the heading change desired.

Given this idealized control law, the corresponding state component time histories $\underline{x}_C(\tau)$ for $0< \tau = t - t_{Trig} < (\tau_M + \Delta\tau_D)$ can be computed according to a good dynamical model; the additional time period $\Delta\tau_D$ at the end is added because in real dynamical maneuvers the transition is not completed at the time when the feed-forward control input ends. In order to counteract disturbances during the maneuver, the difference $\underline{\Delta x}(\tau) = \underline{x}_C(\tau) - \underline{x}(\tau)$ may be used in a superimposed state feedback controller to force the real trajectory towards the ideal one.

The general state feedback control law is

$$\underline{u}(\tau) = -K^T \underline{\Delta x}(\tau), \qquad (5)$$

with K an r by n gain matrix. The gain coefficients may be set by pole placement or by a Riccati design (optimal linear quadratic controller) well known in control engineering (Ref. 17). Both methods include knowledge about behavioral characteristics along the time axis: While pole placement specifies the eigenvalues of the closed loop system, the Riccati design minimizes weighted integrals of state errors and control inputs.

The simultaneous use of dynamical models for both perception and control and for the evaluation process leading to behavior decision makes this approach so efficient. Figure 4 shows the closed-loop interactions in the overall system.

Based on object state estimation (lower left corner) events are detected (center left) and the overall situation is assessed (upper left). Initially, the upper level has to decide which of the behavioral capabilities available are to be used: Feed-forward, feedback, or a superposition of both; later-on, the feedback loops activated are running continuously (lower part in fig. 4) without intervention from the upper levels, except for mode changes. Certain events also may trigger feed-forward control outputs directly (center right).

Since the actual trajectory evolving from this control input may be different from the nominal one expected due to unforeseen perturbations, commanded state time histories $\underline{x}_C(\tau)$ are generated in the block object 'state prediction' (center of fig. 4, upper right, central block in depth direction) and used as reference values for the feedback loop (arrow from top at lower right from center). In this way, combining feed-forward direct control and actual error feedback, the system will realize the commanded behavior as close as possible and deal with perturbations without the need for re-planning on the higher levels.

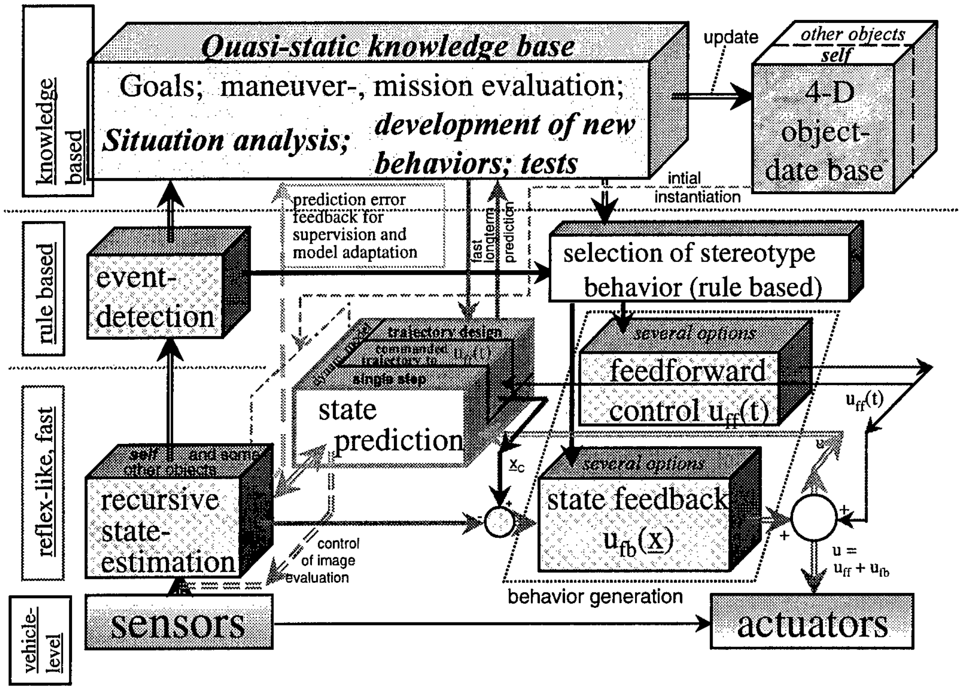


Figure 4: Knowledge based real-time control system with three hierarchical levels and time-horizons.

All, that is needed for mission performance of any specific system then is a sufficiently rich set of feed-forward and feedback behavioral capabilities. These have to be activated in the right sequence such that the goals are achieved in the end. For this purpose, the effect of each behavioral capability has to be represented on the upper decision level by global descriptions of their effects:

1. For feed-forward behaviors with corrective feedback superimposed (case 3 given above) it is sufficient to just represent initial and final conditions including time needed; note that this is a quasi-static description as used in AI-methods. This level does not have to worry about real-time dynamics, being taken care off by the lower levels. It just has to know in which situations these behavioral capabilities may be activated with which parameter set.

2. For feedback behaviors it is sufficient to know when this mode may be used; these reflex-like fast reactions may run over unlimited periods of time if not interrupted by some special event. Typical examples are heading control or altitude keeping in air vehicle guidance; the integral of speed then is the distance traveled. These values are given in information systems for planning, like maps or tables, and can be used for checking mission progress on the upper level.

Performing more complex missions on this basis has just begun. The newly available computing power will lead to quick progress on this mission level with the general concept being well defined.

3.2 Multiple loops in dynamic scene understanding

The principles discussed above have lead to parallel realizations of multiple loops in the interpretation process both in space and in time; figure 2 has displayed the spatial aspects. In the upper half of the figure, the essential scales for feedback loops are the object level, the local situation level, and the global mission performance level on which behavior decisions for achieving mission goals are being done.

These decisions may be based on both local and extended predictions of the actual situation and on knowledge about behavioral capabilities of the own vehicle and of other subjects in the scene. The multiple loops used in our system in the time domain are displayed in figure 5; they range from the millisecond scale for inertial viewing direction control to several hours for ground and flight vehicles on the mission scale encompassing sequences of maneuvers and feedback behavioral modes.

Up to now, the outermost two loops labeled 'quasi-static' are closed mainly by human operators and software developers. They are tackled now for automation according to the system structure developed above; a unified approach encompassing techniques from systems dynamics, control engineering, computer simulation and animation as well as methods from AI has become feasible.

More aspects of the cooperation between hierarchical levels for perception and control will be discussed in the section on system integration below.

4. INDIVIDUAL ASPECTS OF SITUATIONS

The same arrangement of objects and subjects may be interpreted as a different situation for different subjects according to their individual goals they are striving for. In addition, looked at from a higher hierarchical level this same geometrical arrangement may be interpreted as again a different situation in the framework of the overall evaluation in the mission context.

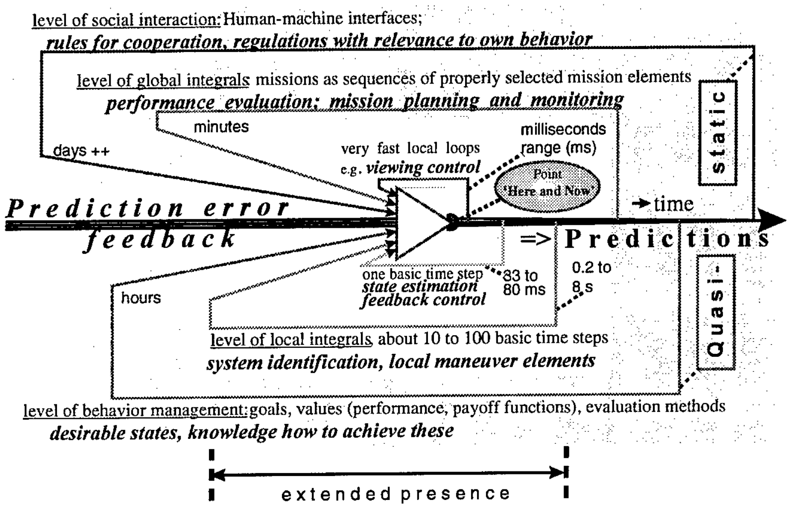

level of social interaction: Human-machine interfaces;
rules for cooperation, regulations with relevance to own behavior
level of global integrals missions as sequences of properly selected mission elements
performance evaluation; mission planning and monitoring
minutes
milliseconds range (ms)
days ++
very fast local loops e.g. viewing control
Point 'Here and Now'
→time
Prediction error feedback
=> Predictions
static
Quasi-
one basic time step
state estimation
feedback control
33 to 80 ms
0.2 to 8 s
hours
level of local integrals, about 10 to 100 basic time steps
system identification, local maneuver elements
level of behavior management: goals, values (performance, payoff functions), evaluation methods
desirable states, knowledge how to achieve these
extended presence

Figure 5: Multiple feedback loops on different time scales in the 4-D approach and corresponding representational levels

Situations are specific to the perceptual capabilities of acting subjects in the actual task context and to decision making with respect to the behavioral capabilities for achieving mission goals and avoiding safety hazards. In figure 6 a scheme is given for monitoring how the actual situation is embedded in a mission and a maneuver context; scaling each unit on one scale to the range 0 – 1 allows easy checks on the different levels. Usually, individual side constraints and value systems are taken into account on different levels. On the mission level, the effect on the final performance measure may dominate, while on the maneuver level local aspects like safety margins are of higher concern. In order to arrive at flexible yet autonomous robot systems consisting of several units, each individual unit should not just have its 'personal' value system but should also have knowledge about the value systems of the higher hierarchical levels. Parallel evaluations of any one given situation according to several of these systems and comparing the relative benefits for oneself and the group or the larger unit one belongs to, will allow rational behavioral decisions meriting the label intelligent.

This means that the introduction of a hierarchy of goal functions and value systems is required for creating the capability of complex responsible decision making in an autonomous system. Only by weighting the different results correspondingly, can intelligent behavior be generated. Decisions should not be taken according to some preprogrammed priority scheme but with respect to a value system in order to achieve really intelligent behavior.

Therefore, perceiving all the different process components and subjects involved in a situation in parallel is necessary for arriving at intelligent autonomous actions. On the different spatial and temporal scales specific situation monitoring is required; only the sense of vision is sufficiently rich in providing information with respect to the different aspects involved. The spatial scales are taken care of by multi-focal resolution of the sensor arrangement in the **MarVEye**-concept. By evaluation of a situation from near to far, active viewing direction control can direct the relatively small area of high resolution (field of view of the tele-cameras) to the areas of most interest in the real world. New events may thus be detected efficiently; a change in situation will then trigger proper behavioral reactions.

In a nap-of-the-Earth flight of a helicopter, the inertially stabilized camera platform may lock onto a well discernible feature grouping at the horizon while the image flow of features from obstacles nearby is being roughly tracked in the images of the wide-angle cameras. Should a potentially critical obstacle occur, or would the mission require a maneuver, different perceptual and behavioral modes would be activated. For example, viewing direction control could switch to fixation of some group of features on the most safety critical obstacle (stationary or moving) while other objects or subjects of less importance actually would only be roughly tracked in one of the cameras with a wider field of view.

The temporal scales of interest on the different levels are realized by corresponding representations according to figures 5 and 6. All these activities run in parallel on a distributed processor system with message passing as the means for communication.

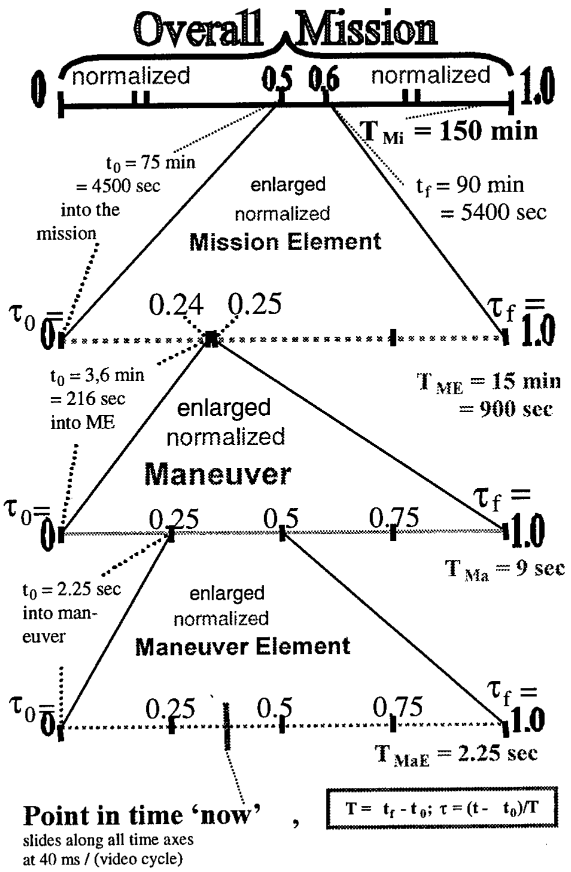

Overall Mission
0
normalized
0.5 0.6
normalized
1.0
T Mi = 150 min
t0 = 75 min = 4500 sec into the mission
enlarged normalized Mission Element
tf = 90 min = 5400 sec
τ0 = 0
0.24 0.25
τf = 1.0
t0 = 3,6 min = 216 sec into ME
T ME = 15 min = 900 sec
enlarged normalized Maneuver
τ0 = 0
0.25
0.5
0.75
τf = 1.0
T Ma = 9 sec
t0 = 2.25 sec into maneuver
enlarged normalized Maneuver Element
τ0 = 0
0.25
0.5
0.75
τf = 1.0
T MaE = 2.25 sec
Point in time 'now'
slides along all time axes at 40 ms / (video cycle)
T = tf - t0; τ = (t - t0)/T

Figure 6: Multiple scales representation of time t for mission performance at four temporal levels

Thus, situations are defined according to own perceptual and behavioral capabilities as well as the representational scheme realized for making use of those resources.

5. THE INDIVIDUAL SITUATION AS PART OF TASK EMBEDDING

As already mentioned above, in more complex environments for autonomous systems there is not just one individual autonomous system acting on its own; usually, the individual system is part of a society of acting agents, both natural (living beings) and technical ones. Of special interest are those 'friends' striving for the same goal; in addition, a distinction between subjects in the outside world who are merely passive with respect to the own action and those who probably have conflicting goals may be important for situation assessment. For the latter ones it is especially useful to know what these agents will be able of doing in certain situations and how likely it is that they will really act in such a manner.

On a higher level of a hierarchical system, one agent with an 'overview' from an elevated platform may get information about several agents, both friendly and other ones. He may be able to come up with behavioral decisions superior to those derived from local information and limited horizons for visual observation. Besides, tactical or strategic considerations may lead him to follow a different overall plan. Sometimes, cooperative intelligent agents with the capability of information exchange, be it modern telecommunication or archaic visual signs (like hand waving or similar), may gain advantage of coordinating their efforts as well as possible. This can be achieved by 'internalizing' the decision making process usually done on the next higher hierarchical system level through 'imagining' the situation as seen from the higher point of view (see dashed parabolic line in figure 7). Of course, the information available to the individual agent on the lower level is that which he can reasonably infer from the situation as perceived personally plus those components communicated or derived from knowledge and observed indications by vision.
In this process of situation assessment, the own role is analyzed 'as seen from above'. Cooperative efforts need this additional step for situation evaluation with goal functions of the next higher level(s); this type of organization for arriving at decisions quite naturally leads to the question of priority of higher level goals over individual goals? In the long run, problems like those of morale in human societies may occur for technical systems also.

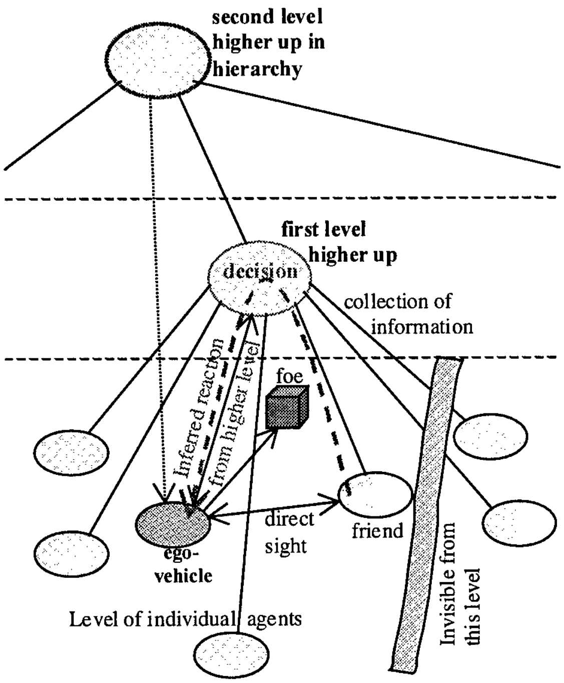


Figure 7: Evaluation of a complex situation exploiting visual sensing and background knowledge on cooperative behavior

Only the sense of vision in connection with background knowledge on task related behaviors of subjects allows to remotely collect and intelligently interpret all the information needed on other agents in order to take advantage of the environmental situation encountered. A unified representational scheme has been presented with the 'scene tree' in (Ref. 18).

6. SYSTEM INTEGRATION ASPECTS

Contrary to conventional measurements, there is no direct contact with an object observed in visual perception; all properties of the objects are just hypothesized and claimed to be true. Hypothesis generation on the base of visual features consistently detected is the main source of knowledge about other objects and processes in the environment. Note that it is the combination of visually gained data with background knowledge from previous experience, which allows the system to perceive 'the environment'. Therefore, hypotheses once started have to be checked every now and then in order not to be stuck with percepts poorly supported by the visual data stream.

Temporal predictions on different scales and their critical checks will allow discovering inconsistencies. The availability of a diverse sensor array in the complex vehicle eye **MarVEye** may also contribute to avoiding confusion of features, one of the major sources of misperceptions. In addition, checking the output of a diverse sensor array like vision and inertial sensors on the own body lends itself for the detection of some kinds of inconsistencies.

Figure 8 shows a survey of overall system integration with five levels:

- The vehicle with its sensors and actuators forms the lowest (hardware) level.
- Signal processing of raw sensor data and for control output makes up level two.
- On top of this is the so-called 4-D level with specialists for object hypothesis generation and for continued perception of
    - o the inertial ego-state (IbSE)
    - o 3-D ground surface profiles (vertical structure of the environment, 3DS)
    - o road networks on the surface (RDT),
    - o stationary landmarks (LDT) for visual navigation, and
    - o mobile objects (ODT).

  A 'temporal prediction' component supports recursive estimation by single step predictions, maneuver realization by the generation of short-term reference trajectories, and situation assessment for decision making by longer-term, fast in-advance simulations for the most safety-critical objects and subjects in the vicinity. Feed-forward and feedback control computations for the generation of behaviors (with several options for different situations and tasks) is shown to the right on this level.

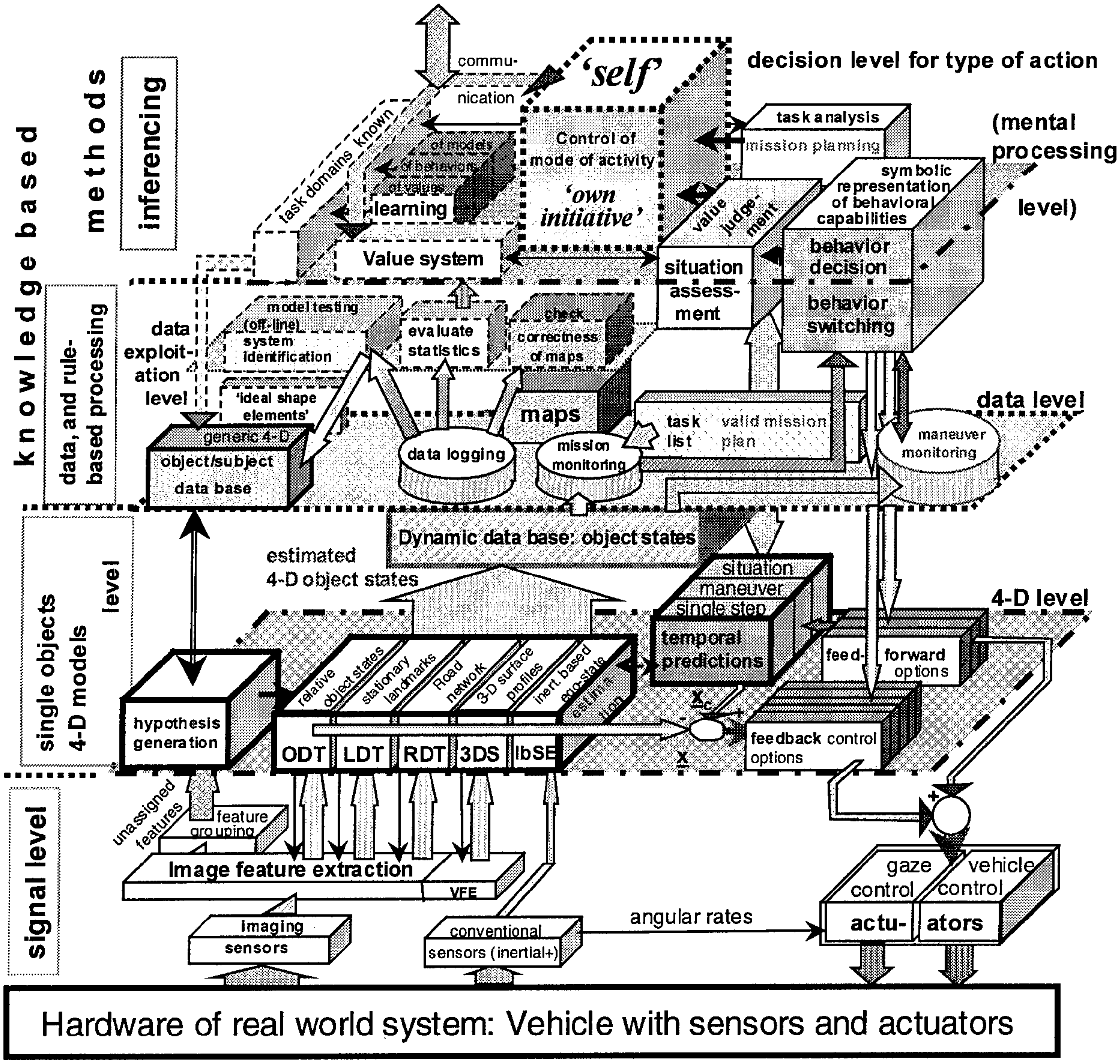


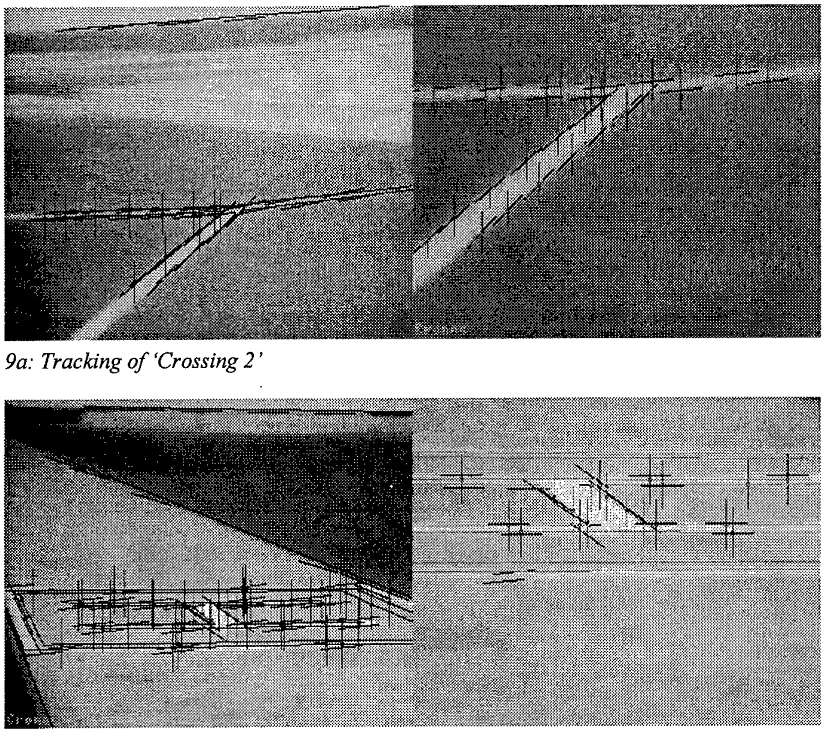
Figure 8: Overall system integration for visual perception in a closed action loop with the real world; through this loop closure, predictions based on models of dynamical processes may be checked, and the prediction errors are an indication of the quality of internal representations. Solid blocks have been tested up to now; dotted ones are components under development for the third generation of software implementation of the 4-D approach.

- As second level from the top, the 'object data level' is shown. It encompasses the 'generic 4-D object data base' (background knowledge specific to certain task domains), the dynamic data base containing all the best estimates for the relative states of all objects actually instantiated, and stored data of the mission plan and the map information needed. Logging of actual data is performed here, too; these data will be used in the future on an intermediate data exploitation level for keeping statistics on objects and situations encountered, for learning from experience (system identification) and for checking the consistency of the value system.

- The top level of 'mental processing' has just started emerging from situation assessment and behavior decision. It is intended to become the center for high-level decisions and the one and only instance for communication with the human operator. Here, responsibility for actions undertaken will be residing, and adaptation of the value system for better overall consistency will be performed. This will be a construction site for years to come.

7. EXPERIMENTAL RESULTS

In preparation of real mission performance with a helicopter, a small mission with landmark navigation in the vicinity of the airport of Brunswick, Germany, has been studied in a hardware-in-the-loop simulation with real-time image sequence processing active. The performance achieved can best be assessed from the video documentation made. Figure 9 shows a scene with a road junction tracked as a landmark and as a way-point for starting a new leg of the mission plan. The two images are those taken with the old bifocal camera configuration shown in the upper part of figure 1; viewing



9a: Tracking of 'Crossing 2'

9b: Tracking of taxiways, frame and Heli H during final approach

Figure 9: Visual landmark navigation of an autonomous helicopter using a road junction as a way-point for mission performance. Top: Image of camera with mild tele-lens; bottom: same scene with stronger tele-lens for improved spatial resolution (foveal vision) .

range is the same in the left and right image. Focal lengths of the cameras differ by a factor of 2. In fig. 9a left the horizon is tracked beside the road junction, while the right (foveal) image of the strong tele-camera concentrates on the road junction. The helicopter finally stops hovering above a spot marked by the letter (capital) H on the ground as used for designating the landing area for helicopters (see fig. 9b).

8. CONCLUSIONS

It is the combination of visual and inertial sensing in the task context of controlling ego-motion which allows a subject to develop intelligence. In order to be able to interpret both data streams consistently, temporal representations of the process variables have to be developed. The most economical way to achieve this is to use generic differential models containing both state and control variables, and to make use of the operation of integration over time. These models are used both for perception (state estimation in 3-D space) and for control actuation (feedback for counteraction of perturbations and feed-forward for fast maneuver control). These control activities depend on the situation encountered; recognizing situations in a task context and background knowledge about which control activities lead to which results is important for arriving at flexible successful behavior.

With respect to behavior decision for complex cooperative actions it is proposed to evaluate situations not just from an individual point of view, but to also do this with performance criteria usually used on a higher hierarchical level (see Ref. 19 for a similar point of view). This allows for rational decisions well balanced between egoistic and social points of view.

Since all information on other objects based on vision is derived from a combination of remotely measured data and internally stored background knowledge, it is mandatory to steadily check the assumptions underlying the interpretation in order to understand motion processes in the environment correctly and reliably.

Dynamic machine vision is maturing to a state where rather complex scenes and mission elements can be handled. In highway driving, about a dozen vehicles can now be tracked in parallel on three-lane roads. Nap-of-the-Earth flights of helicopters should soon become possible. With a growth rate of computing power of one order of magnitude every 4 to 5 years, autonomous vehicles capable of dynamic vision with performance levels approaching human ones may become feasible in the long run. Even unmanned combat air vehicles with an autonomous sense of vision (UCAV's) seem within reach; their vision sensors need not be confined to the optical range. Imaging radar as well as infra-red sensing is available. It is the right time now to start studying this new technology with vision as a key component in more depth.

9. LITERATURE

[1] Dickmanns, E.D.; V. Graefe: a) Dynamic monocular machine vision. Machine Vision and Applications, Springer International, Vol.1, 1988, pp 223-240. b) Applications of dynamic monocular machine vision. (ibid), 1988, pp 241-261.

[2] Dickmanns, E.D.: Vehicles Capable of Dynamic Vision. 15th International Joint Conference on Artificial Intelligence (IJCAI-97), Nagoya, Japan, August 23-29, 1997.

[3] Dickmanns, E.D.; R.F. Schell.: 'Visual Autonomous Automatic Landing of Airplanes'. AGARD Symp. on 'Advances in Guidance and Control of Precision Guided Weapons', Ottawa, Canada, May 1992

[4] Wuensche, H.-J.: Detection and Control of Mobile Robot Motion by Real-Time Computer Vision. In N. Marquino (ed): Advances in Intelligent Robotics Systems. *Proceedings of the SPIE*, Vol. 727, 1986, pp 100-109.

[5] Dickmanns, E.D.: Machine Perception Exploiting High-Level Spatio-Temporal Models. *AGARD Lecture Series 185* 'Machine Perception', Hampton, VA, Munich, Madrid, Sept./Oct. 1992.

[6] Dickmanns, E.D.: Subject-Object Discrimination in 4-D Dynamic Scene Interpretation Machine Vision. Proc. IEEE-Workshop on Visual Motion, Newport Beach, 1989, pp 298-304

[7] Dickmanns, E.D.: Road vehicle eyes for high precision navigation. In Linkwitz et al. (eds): *High Precision Navigation.* Dümmler Verlag, Bonn, 1995, pp. 329-336.

[8] Luenberger, D.G.: Observing the state of a linear system. *IEEE Trans on Mil. Electronics* 8, 1964, pp 290-293.

[9] Meissner, H.G.; E.D. Dickmanns: Control of an Unstable Plant by Computer Vision. In T.S. Huang (ed): Image Sequence Processing and Dynamic Scene Analysis. Springer-Verlag, Berlin, 1983, pp 532-548.

[10] Wuensche:, H.-J. Verbesserte Regelung eines dynamischen Systems durch Auswertung redundanter Sichtinformation unter Berücksichtigung der Einflüsse verschiedener Zustandsschätzer und Abtastzeiten. Report HSBw/LRT/WE 13a/IB/83-2, 1983.

[11] Werner, S.; S. Fürst; D. Dickmanns; E.D. Dickmanns: A vision-based multi-sensor machine perception system for autonomous aircraft landing approach. *Enhanced and Synthetic Vision, AeroSense '96,* Orlando, FL, April 1996.

[12] Werner, S.: Maschinelle Wahrnehmung für den bordautonomen automatischen Hubschauberflug. PhD thesis, UniBwM, LRT, 1997.

[13] Schiehlen, J.; Dickmanns E.D.: Two-Axis Camera Platform for Machine Vision. AGARD Conference Proc. 539 'Pointing and Tracking Systems', Seattle 1993, pp 22-1 - 22-6

[14] Hock, C.: Wissensbasierte Fahrzeugführung mit Landmarken für autonome Roboter. PhD thesis, UniBwM, LRT, 1994.

[15] Thomanek, F.: Visuelle Erkennung und Zustandsschätzung von mehreren Straßenfahrzeugen zur autonomen Fahrzeugführung. PhD thesis, UniBwM, LRT, 1996.

[16] Müller, N.: Autonomes Manövrieren und Navigieren mit einem sehenden Straßenfahrzeug. PhD thesis, UniBwM, LRT, 1996.

[17] Kailath, T.: Linear Systems. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1980.

[18] Dickmanns, D.: Rahmensystem für visuelle Wahrnehmung veränderlicher Szenen durch Computer. PhD thesis, UniBwM, Informatik, 1997.

[19] Albus, J.: Outline for a Theory of Intelligence. IEEE Trans. Systems, Man, and Cybernetics, Vol. 21, No. 3, May/June 1991.

Advances in Soft-Computing Technologies and Application in Mission Systems

U. Krogmann
Bodenseewerk Gerätetechnik GmbH
Postfach 10 11 55
D-88641 Überlingen

The objective of this paper is to give a short report about the AGARD Lecture Series 210 with the above title.

Tactical systems are implemented as Integrated Mission Systems (IMS) such as air and space defence systems. As shown in the lower part of Fig. 1, the key elements of an IMS are platforms with sensors and effectors, ground based components with communication, command and control etc.. Mission management constitutes the functional process within and among Integrated Mission Systems (Fig. 1). It will provide the capability for rapidly gathering, distributing and integrating large quantities of available information and will allow rapid strategic and tactical decision-making on the missions as well as carrying out the resulting actions from what has been decided and what is required for dealing effectively with these missions, in order to perform the functionalities appropriately in unpredictable and uncertain scenarios.

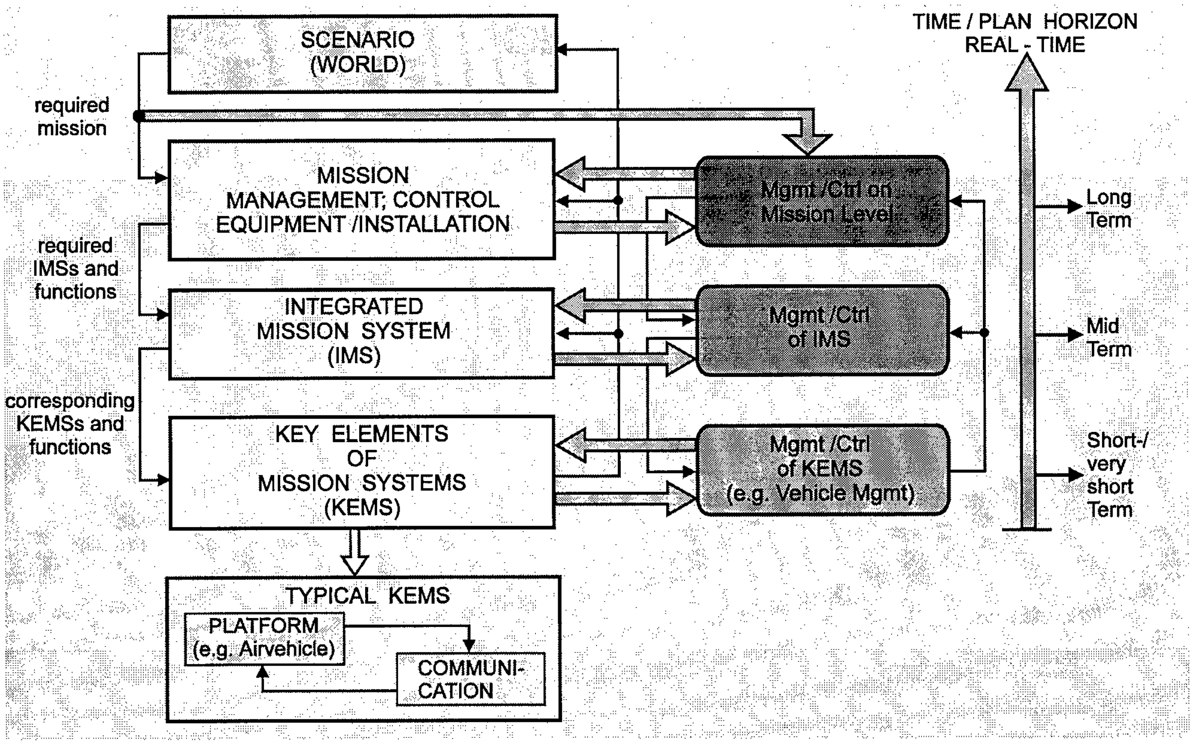

SCENARIO (WORLD)
required mission
MISSION MANAGEMENT; CONTROL EQUIPMENT /INSTALLATION
required IMSs and functions
INTEGRATED MISSION SYSTEM (IMS)
corresponding KEMSs and functions
KEY ELEMENTS OF MISSION SYSTEMS (KEMS)
TYPICAL KEMS
PLATFORM (e.g. Airvehicle)
COMMUNI-CATION
Mgmt /Ctrl on Mission Level
Mgmt /Ctrl of IMS
Mgmt /Ctrl of KEMS (e.g. Vehicle Mgmt)
TIME / PLAN HORIZON
REAL - TIME
Long Term
Mid Term
Short-/ very short Term

***Figure 1:* Mission system and mission management structure**

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

Taking for example an airvehicle as a mission system element, Fig. 2 depicts how mission control interacts with the functional levels of the guidance and control (G.a.C.) process of the airvehicle. It can be seen that G.a.C. problems extend over several hierarchically structured levels and the communication functions between these levels. The represented interconnection of the different functional levels (scenario, mission, trajectory, airvehicle state) can be conceived of as a hierarchically structured control system. The objects on which G.a.C. functions are performed on the mentioned levels represent the control plants. Information processing by which actuation is generated from sensor information on all levels represents the controller functions which are often also called the recognize-act-cycle functions.

It requires functions such as recognizing and assessing the situation, defining action goals, generating optimum or favorable solutions, decision-making, planning and finally performing as well as monitoring of actions. Hence, behavior levels of mental capabilities such as skill, rule, knowledge based functions can be assigned to the functional levels.

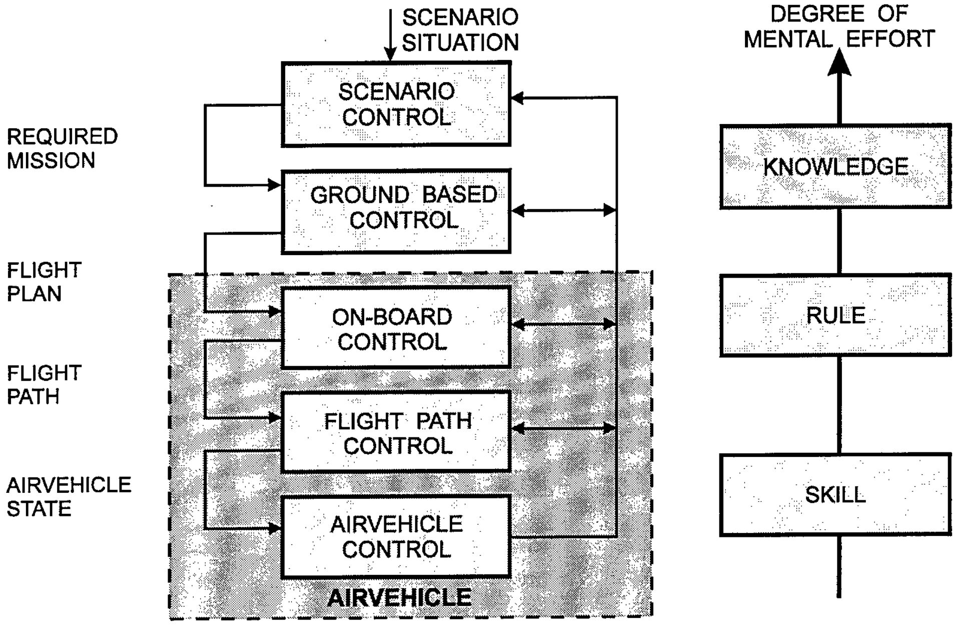


Figure 2: ***Cascaded airvehicle G.a.C. structure***

For reasons of human limitations in more demanding dynamic scenarios and in the operation of complex, highly integrated systems, there is the neccessity for extended automation of these functions on higher levels such as trajectory control as well as mission management and control. Furthermore, the implementation of intelligent functions on lower levels such as the fusion and interpretation of sensor data, multifunctional use of sensor information and advanced nonlinear learning control becomes inevitable.

This is accomplished through the introduction of new computational and machine intelligence techniques (CMI). Moreover, CMI will be the most important prerequisite for less manned air operations extending as far as fully autonomous tactical platforms.

Although the development of hardware and software for computers of conventional v. Neumann architecture has continued for more than 20 years and the performance of today´s processors is 25.000 times better than in the 1970s, the dynamics of this development is going on as well.

However, there is a complementary shift from conventional computing techniques, including symbolic AI/KB techniques, to so-called soft computing technologies. The new paradigm is based on modelling the unconscious, cognitive and reflexive function of the biological brain. This is accomplished by massively parallel implementation in networks as compared to program/software based information processing in conventional sequential architectures.

In contrast to the conventional method, soft computing addresses the pervasive imprecision of the real world. This is obtained by consideration of the tolerances for imprecision, uncertainty and partial truth to achieve tractable, robust and low-cost solutions for complex problems.

Important related computing methodologies and technologies include among others fuzzy logic, neuro-computing, as well as evolutionary and genetic algorithms. With those a viable step towards intelligent machines can be expected that offer autonomous knowledge acquisition and processing, self-organization and structuring as well as associative rule generation for goal-oriented behavior in rarely predictable scenarios. The new techniques will yield computational and machine intelligence which offers the user the opportunity for cognitive automation of typical „recognition-act cycle" activities on various functional and operational levels.

The last two decades have witnessed a very strong growth of CMI techniques. These techniques have already been applied to a variety of problems to deliver efficient solutions to the benefits of the user. Certainly there are relationships between CMI and other fields such as those shown on top of Fig. 3. Moreover, numerous disciplines have contributed to the area of soft computing where some are mentioned at the bottom of Fig. 3.

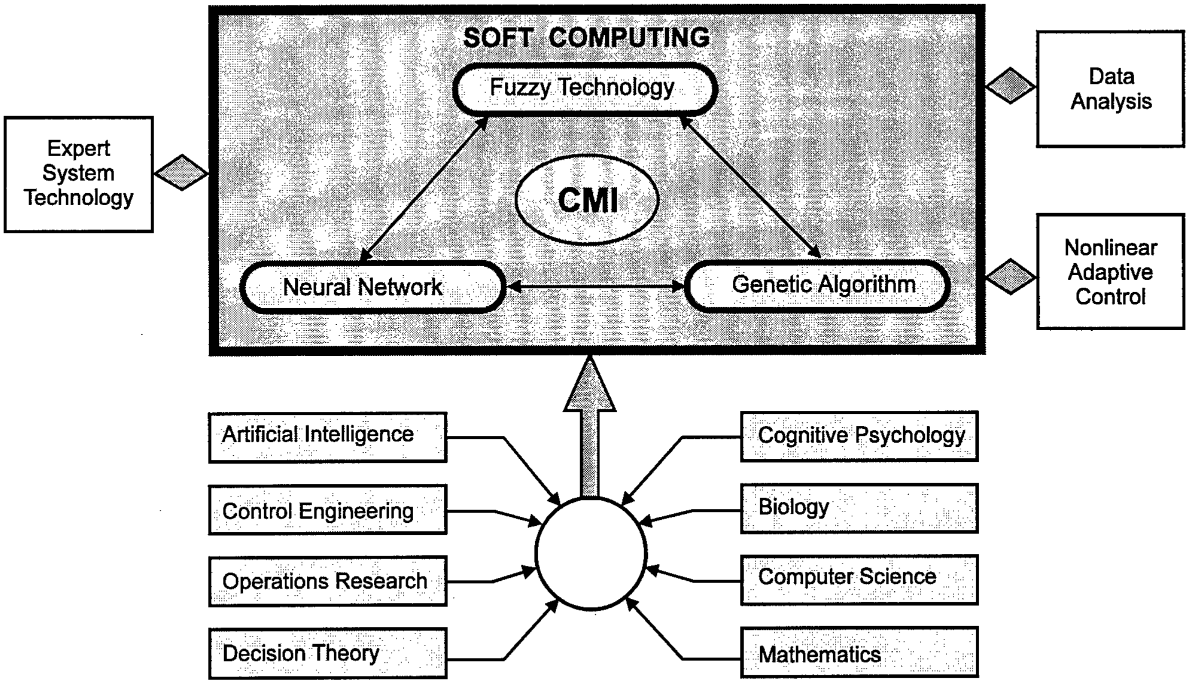


Figure 3: Relationships to and contributions from other areas and soft computing/CMI

The Mission Systems Panel (MSP) of AGARD felt it particularly important and timely to facilitate, foster and strengthen communication and cooperation between scientists, practitioners and decision-makers in various disciplines.

In this context a Lecture Series was seen as a valuable step where the objective was to introduce soft computing as the basis for CMI and to briefly familiarize the participants with important related technologies and techniques as well as applications. Under the sponsorship of the MSP and the Consultant and Exchange Programme of AGARD the Lecture Series was organized and presented during September and October 1997 in Canada, The Netherlands, Spain and Turkey. The material is published in the document AGARD LS-210. A short survey regarding the contents of the lectures is given as follows.

The first lecture introduces and overviews the soft computing techniques enabling CMI. Based on brainlike structures the most important techniques of soft computing such as fuzzy logic, artificial neural networks and genetic algorithms are briefly described. The treated approaches form the basis for adaptive, learning control, as well as advanced automation and artificially intelligent machines.

The series continues with an introduction and review of neural networks, their basic theory and important network models. It also looks at the application potential for mission systems.

Fundamentals of fuzzy logic, fuzzy inference and fuzzy control are treated in the third lecture, which is followed by a presentation of the basics of genetic algorithms and evolutionary computing.

A paper on hybrid architectures for intelligent systems concludes the part which is dealing with fundamental aspects. Various ways of combining fuzzy and neural elements for classification and control applications are presented.

The second part of the lecture series is devoted to applications of soft computing technologies in mission systems. Contributions address sensor signal processing, guidance and control problems, mission management and simulation issues as well as autonomous systems.

The first lecture demonstrates a combination of Kalman filter and maximum likelyhood techniques with neural networks for acquiring and tracking targets.

It is followed by a paper dealing with the application of neural networks and fuzzy techniques in the area of guidance and control. The emphasis is on reconfiguration of damaged aircraft.

The last paper but one considers the application of genetic algorithms and evolutionary strategies for mission management, simulation and autonomous systems.

Finally, the last paper summarizes important enabling techniques and technologies for the implementation of future autonomous systems. Main emphasis is placed on information technology with its soft computing techniques.

During the round table discussion at the end of the series the following aspects were addressed.

Where are we now?

- Proven paradigms and techniques.
- What systems have been fielded?
- Proven applications?
- Hardware capabilities: What will current hardware systems yield in functionality and performance?
- Neural, fuzzy and genetic engineering: How to design, develop, manage, document, maintain successful systems?
- Hybrid systems design; integration issues; systems with algorithmic software, expert systems, ANNs and fuzzy calculus (software and hardware)

Where are we going?

- Future trends:
- Where are we likely to be going at the end of the century and what is required to exploit the new technological potential?

How do we get there?

- More R&D, project design methodology, from knowledge to behavior engineering; QA and documentation, standards, needed tools.

How long does it take?

- 10 years? Things that are rather well developed.
- 20 years? Activities that are still in universities, but are highly developed.
- 30 years? Work that is just beginning in university research.

An excerp from the application potential covers among other things

- Data compression
- Pattern recognition and classification
- Image processing
- System identification
- Intelligent control
- Signal processing
- Real-time optimization
- Feature extraction
- Time series prediction
- Function approximation
- Memory recall
- Various cognitive tasks

Finally the Lecture Series concluded with the following main remarks:

Fuzzy and artificial neural network techniques enable the endomorphic modelling of real world objects and scenarios. Together with conventional algorithmic processing, classical expert systems, probabilistic reasoning techniques and evolving chaos-theoretic approaches, they enable the implementation of recognize-act cycle functions.

Genetic and evolutionary algorithms can be applied to generate and optimize appropriate structures and/or parameters to acquire, encode, represent, store, process and recall knowledge.

This yields self-learning control structures for dynamic scenarios that evolve, learn from experience and improve automatically in uncertain environment. Ideally, they can be mechanized by a synergetic, complementary integration of fuzzy, neuro- and genetic techniques.

Thus soft-computing techniques support the move towards adaptive knowledge based systems which rely heavily on experience rather than on the ability of experts to describe the dynamic, uncertain world perfectly.

Image Data Fusion for Enhanced Situation Awareness

H.-U. Döhler, P. Hecker, R. Rodloff
DLR, Institute of Flight Guidance, Lilienthalplatz 7
D-38108 Braunschweig, Germany

1. Introduction.

Today's aircraft crews have to handle more and more complex situations. Especially during approach, landing, take-off and taxiing the improvement of situational awareness can be regarded as a key task.[1] Particularly future military transport aircrafts have to cope with new requirements such as autonomous, non-cooperative landing at unsupported (without D-GPS, MLS, ILS) airstrips down to CAT-I (minimum) or better, low level flight operations, ground mapping, precise air dropping, search and rescue missions. In most cases these requirements have to be established under adverse weather conditions, without being detected by hostile observation.

Within this context visual information provided by fusion of image data from onboard multispectral sensors with synthetic vision (SV), supported by an ATC - interface and aircraft state data (position, attitude and speed), will become an important and helpful tool for aircraft guidance. The development of these so called "Enhanced Vision Systems" (EVS) is an interdisciplinary task, which requires a wide spectrum of different information technologies:

- modern data link technology for transmission of guidance information;
- complex data bases to provide terrain data for synthetic images and to support the imaging sensors (sensor characteristics, object signatures);
- high performance computer graphics systems to render synthetic images in real time;
- a new generation of onboard imaging sensors, like solid state infrared and especially a new kind of imaging radar, providing a real view through darkness and adverse weather;
- knowledge based image interpreters to convert sensor images into a symbolic description.

This paper presents the DLR concept for an integrated enhanced vision system. After the description of the basics in section 2 the image sensor characteristics are compared with the requirements of an Enhanced-Vision-System in section 3. First results concerning the experiments with the DASA HiVision radar and data fusion techniques are given in section 4.2 and 4.3.

2. The DLR concept for enhanced vision.

Looking a little closer to the meaning of the terms "Enhanced Vision Systems" (EVS) and "Synthetic Vision System" (SVS), we find a rather indistinct situation: sometimes the whole field concerning "obstacle detection", "sensor simulation", "4D flight guidance display", "enhanced vision", etc. is subsumed under the headline "Synthetic Vision" /2,3/ and sometimes the same term is strictly reduced to computer generated images /1/. But as an average one finds the following use of these two terms:

It's amazing to see that the terms "EVS" and "SVS" are very often discussed in parallel, - sometimes even treated as competing concepts. But as long as EVS and SVS are considered as isolated technologies, they have to face some critical questions, such as:

"EVS" - no single sensor will really cover all possible weather situations;

- what happens if the imaging sensor fails?

- multispectral images are difficult - sometimes impossible - to interpret.

"SVS" - unmodelled obstacles are not detectable;

- how can the integrity of the data base be monitored and what happens if data base errors occur?

- how can such a data base be certified?

- position accuracy of the synthetic image depends on the accuracy of the reference system .

Another somewhat restricted but quite interesting definition concerning the field of EVS / SVS - terminology was coined by the Air Transport Association (ATA) : ... "a means to safely increase airport capacity and reduce runway incursions in low visibility conditions, without significant expansion of ground facilities". This definition seems to be mainly influenced by civil needs, but the renunciation of supporting ground facilities makes EVS- and SVS-technology also very interesting for military requirements.

[1] Recommendation of the FAA - Human Factors Team (SA-1): "The FAA should require operators to increase flightcrews understanding of and sensitivity to maintaining situation awareness, particularly: ...... Position awareness with respect to the intended flight path and proximity to terrain, obstacles or traffic ;".

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

To avoid the drawbacks of the isolated EVS and SVS technologies and to maintain the benefits of both, it seems to be absolutely necessary to integrate them into one system which should be treated as a part of an artificial pilot assistant.

In order to avoid any confusion concerning the terms used throughout this paper, it seems to be worthwhile to define them :

"Sensor Vision" = Sensor generated images; (replaces the term "Enhanced Vision" in the conventional meaning).

"*Synthetic Vision*" = Computer generated images based on navigation inputs, map informations and/or terrain data bases

"Enhanced Vision" = A concept which integrates "Sensor Vision", "Synthetic Vision", aircraft state - and ATC - data .

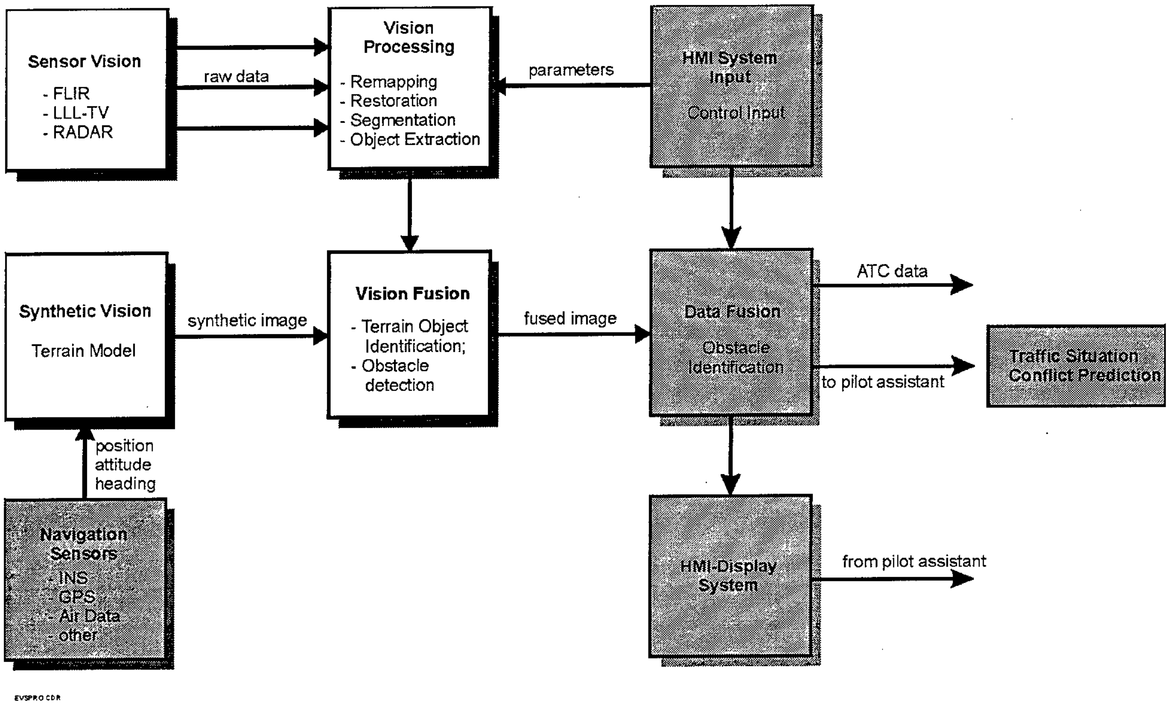


Fig. 2.1: Block Diagram of the DLR "Integrated Enhanced Vision System".

Figure 2.1 shows the basic idea of an integrated Enhanced Vision concept. Key elements are functional subblocks which are interacting through defined interfaces: The "Sensor Vision" - block includes the imaging sensor(s) and possibly some image data preprocessing. The "Vision Processing" - block is responsible for a first step of image processing concerning signal restoration like noise reduction and various filtering techniques, scaling and transformation to convenient perspectives and as a most challenging topic: feature - or object extraction. The block "Synthetic Vision" represents all necessary technologies to set up a synthetic image derived from terrain data bases and navigation informations.

The core part of the integrated Enhanced Vision System is represented by the block "Vision Fusion". A first and simple possibility for "Vision Fusion" could be an integration of sensor images and flight status data on the primary flight display, using a Head Down Display (HDD) or a Head Up Display (HUD)in order to increase the crew's situation awareness, - but other possibilities are just as important:

- incompatibilities between sensed data and terrain data can be used for obstacle detection;
- a conformity check between sensed vision and synthetic vision can be used as an "Integrity Monitor" for the navigation system and the data bases.

The shaded blocks of Figure 2.1 are part of, or connections to a cockpit interface and / or a pilot assistant system, which will not be discussed in this paper. But it is obvious that obstacle detection carried out within the "Vision Fusion" subblock can be improved to an obstacle - or vehicle identification, if ATC informations are included. The pilot assistant may condense these data to a representation of the traffic situation and - if the need arises - to a conflict prediction.

	Advantage :	Disadvantage :
Synthetic Vision	easy image interpretation	no obstacle detection reference system necessary depends on data base reliability
+ **Sensor Vision**	obstacle detection no reference system necessary no data base necessary	difficult image interpretation
+ **Aircraft status**	already available	single sensor information
+ **ATC data**		Needs data link technology
= **Integrated Enhanced Vision System**		

Table 2.1: Definition of the Integrated Enhanced Vision System.

Table 2.1 summarizes the concept of an integrated Enhanced Vision System. It is obvious that the disadvantages of "Synthetic Vision" and "Sensor Vision" cancel each other out and the advantages are increased. In the sense of a synergetic effect both sum up to a new system quality in the fields of

Crew Assistance in connection with:
- adverse weather
- obstacle detection
- non-cooperative landing systems

Integrity Monitor for:
- navigation
- GPWS
- taxi guidance system

Automatic Flight Guidance in terms of :
- image based navigation
- collision avoidance
- terrain avoidance, CFIT,

which would be not achieved if "Sensor Vision" or "Synthetic Vision" are used as isolated technologies.

3. EVS - Sensors : characteristics and requirements.

The requirements for suitable imaging EVS - sensors depend strongly on the application. In the following table the civil requirements and the resulting sensor characteristics should be interpreted as the minimum performance and the military features are meant as an add on.

	operational requirements :	resulting requirements for imaging sensors :
civil :	reduction of minimum approach RVR reduction of minimum takeoff RVR safe taxi operations obstacle warning in critical phases CFIT warning integrity monitor for GPS supported navigation	⇒ weather independent ⇒ sufficient resolution ⇒ image rate > 15 Hz ⇒ image delay < 200 ms ⇒ coverage minimum = HUD coverage
military:	low level flights landing on unsupported forward operating strips precise air dropping air surveillance ground mapping for surveillance search and rescue	⇒ passive or "silent" sensor ⇒ autonomous; ground facilities not necessary ⇒ reconnaissance capability

Table 3.1 : Requirements on imaging EVS - sensor. (RVR = Runway Visual Range; CFIT = Controlled Flight Into Terrain)

With the background of the requirements shown in table 3.1 we should be able to qualify different types of sensors in terms of their EVS - capability. A rather convenient method to distinguish between all possible sensor types, is to refer to the operating wavelength, because this parameter allows a rather easy estimation for two of the most important sensor parameters: the penetration through the atmosphere (especially under foggy conditions) and the image quality in terms of resolution. The rule of thumb is rather simple: increasing wavelength improves the penetration and decreases the image resolution. But of course there is no rule without exception: if the wavelength is much smaller than the water particle size (1 - 10 μm), the penetration increases again. This effect is used for the so called "FogEye" - receiver, which is optimized for UV wavelengths between 0,2 - 0,275 μm/4/. Table 3.2 compares the characteristics of sensors which might come into consideration for EVS - applications.

A valid comparison of the all-weather capability is nearly impossible, because the technical realizations of image generation are too different. First of all, it seems quite clear that the active kind of sensors are more successful penetrating a foggy or rainy atmosphere than the passive ones. "Active" in this context is not only an illuminating kind of device, such as the active mmw – radar, but also those sensors which need an emitting source to overcome the signal to noise barrier, as for instant the UV sensor "Fog Eye" /4/ and to a certain extent the "passive" mmw (PMMW) camera, whose range of visibility can be significantly enlarged, if additional mmw - reflectors are positioned nearby the target. /6/. If we take operational requirements into account we can define:

"Active" sensor systems make use of additional means to increase the emission at or nearby the observed object.

From this point of view the mmw - sensor and the UV- Sensor are purely active, the others are optional active depending on the usage of illumination or emission increasing devices, such as IR- emitting lamps or mmw - reflectors, etc.

sensor type	wave-length (μm)	kind of image	active / passive	ground facility	angular res. (°)	range res. (m)	image rate (Hz)	image delay (sec)	coverage (Azim. x elevation) (°)	visibility (m)
UV /4/	0,2 -0,3	Perspective image	passive	yes 1)	≈0,05	2)	5)	5)	30° x 22°	≈ 800 (at: CAT IIIa conditions)
Video	0,5 - 0,8	Perspective image	passive	no	<≈0,05 4)	2)	> 25	5)	>= 40° x 40°	No improv.
IR /3/	3 - 5	Perspective image	passive	no	<≈0,15 4)	2)	≈ 25	5)	>= 40° x 40°	No improv.
IR	8 -12	Perspective image	passive	no	<≈0,15 4)	2)	≈ 25	5)	>= 40° x 40°	No improv.
MMW 94GHz /3,10/	3190	range angle image	active	(no)	6)	3	10	6)	30° x 22°	6)
MMW 35GHz /3,5/	8570	range angle image	active	(no)	0,25° - 0,8° 7)	6	≈ 15	<0,2	≈ 40° x 28° (range. 10 km)	2500 – 3000 at zero-sight
PMMW /6,7,8/	3190 - 8570	Perspective image	passive	(no)	≈ 0,15 – 0,5 3)	2)	≈17	6)	15° x 17°	≈ 700 at :zero-sight

Table 3.2 : Characteristics of potential EVS - sensors. (UV : Ultra Violet Sensor; IR : InfraRed Sensor; MMW: MilliMeter Wave Sensor; PMMW : Passive MilliMeter Wave sensor)

1) UV - source on the ground necessary;
2) direct range information not available;
3) 1 m antenna square aperture /7/
4) calculated for field of view (FOV) of 40°
5) specific values are not available, but problems are not expected
6) data not available.
7) numerical resolution 0,25°; beamwidth: 0,8°

On the other hand a "passive" sensor is not necessarily the same as a "silent" one. For example: the active mmw – sensor, as it is proposed by DASA Ulm /5/, has a greater range than the hostile ESM detection range. This is an example for an active but silent sensor; and it turns out that the meaning of term "silent" is a matter of mission.

But lets return to the "all weather" capability: A rather convincing method to define the "all weather" characteristic of a sensor was proposed by W.F. Horne e.a. /3/ : these authors measured the received power from the runway and the surrounding grass as a function of distance and various weather conditions. The variation of the received power between grass and runway was used as a measure for the contrast and they defined a 3 dB difference as a threshold for sufficient visibility. The result of these investigations for a 35 GHz mmw –sensor and an IR – sensor is given in table 3.2.

Although the visibility data of the other types of sensors are not completely comparable we can state, that the mmw - sensor seems to be the most successful one.

Another very important feature of the above listed sensors is the type of image generating: UV-, IR-, Video- and PMMW-sensors generate a perspective 2D kind of image, which the human visual-perception-system is evolutionary trained to process into a 3-D-interpretation of the "outside world". The mmw – sensor on the other hand delivers primarily an information about the range and the angular direction of a certain object. This range angle information can be transformed into a view "out-of-the-window", but there is still a lack of information about the objects height or its vertical postion/9/. The presentation of such images needs knowledge about the surrounding elevation, which often is estimated by the so-called "flat-earth-assumption"/12/. On the other side the mmw - sensor delivers a direct information about the distance to certain objects which is extremely valuable for navigation, traffic analysis and obstacle detection and avoidance respectively.

But whatever conclusions are drawn from these different kinds of image generation - it seems to be impossible to claim a general advantage for one of them. In connection with the above used definition of the term "Enhanced Vision" as a fusion of "synthetic vision" and "sensor vision", the type of image generation is an important part of the system philosophy.

On the basis of the sensor requirement (table 3.1) and the sensor characteristics (table 3.2) we should be able to correlate sensor characteristics and requirements. (Table 3.3)

	UV – Sensor [1])	Video - Sensor	IR - Sensor	mmw - Sensor	pmmw - Sensor
weather independent	0	-	-	+	0
sufficient resolution	0	+	+	-	-
image rate > 15 Hz	+	+	+	+	+
image delay < 200 ms	+	+	+	+	
coverage (azimuth x elevation)	0	+	+	+	0
passive sensor	yes	yes	yes	no	yes
"silent" sensor	no/yes	yes	yes	yes	yes
autonomous	no	yes	yes	yes	yes/no
reconnaissance capability	no/yes	yes	yes	yes	yes

[1]) air to ground observation

Table 3.3 : Matrix of EVS-requirements and sensor characteristics. ("+" : good ; "0" : fair; "-" : poor)

Simply counting the "+" and "yes", Table 3.3 gives the impression, that the infrared sensors and the simple video cameras should be the most promising sensors for EVS applications. They offer nearly all features which might be valuable, - except one: the ability to penetrate fog, snow and rain. Walter F. Horne e.a. /3/ stated in their paper "*During approaches and landings in actual weather for 50 foot ceilings and 700-1000 foot visibility the millimeter wave image was not noticeable degraded. During these same in-weather approaches, the IR sensor was not able to provide an image any better than the human eye.*"

If EVS - technology is mainly justified by an increase of the crew's (visual) situation awareness under adverse weather conditions the all weather capabilities of the sensors will become the most important characteristic. From this standpoint of view the UV-, the mmw- and the pmmw-Sensor should be taken into account. The UV- sensor and the passive mmw - sensor need some additional ground facilities (UV-sources and mmw-reflectors), the first one in general and the latter one in adverse weather conditions. /4,6/. Additional ground facilities might be no problem, especially if they are cheap and easy to install, but they restrict the EVS technology to certain scenarios with a ground based infra structure, which might be not always available, especially in military environments.

waveform	FMCW
scanning principle	frequency scanning
centre frequency	35 GHz
scanning (image) rate	16 Hz
emitted power	50/500 mW
resolution (angle)	0.26° (157 pixel/41°)
azimuth beamwidth	0.8°
azimuth coverage	41°
range resolution	6 m (512 pixel /3300m)
Instrumented range	3.5 km
detection range against a person	10 km
weight (radar head)	approx. 15 kg
antenna size	86 x 15 x 30 cm^3

Table 3.4 : Characteristics of the DASA "HiVision" radar.

These were the reasons why DLR decided to use the "HiVision" mm-wave radar from DASA Ulm as the main sensor for EVS research. Technical descriptions are given in several papers /5, 11,/; therefore a short summary might be sufficient: "HiVision" radar uses continous wave (FMCW) technology with a frequency scanning antenna which covers an azimuth sector of 40°. Table 3.4 summarizes the parameters which were achieved during recent tests.

A very promising feature of this sensor, especially for military applications, is the ESM range, where the radar could be detected. With an output power of 100 mW / 500 mW a high performance ESM receiver with a -80 dBm sensitivity would detect the radar within a distance of 2.3 / 5.2 km. On the other hand: the radar range against a person (radar cross section approx. 1 m^2) is 6.9 / 10 km, which means that the hostile detection range is much smaller than the detection range of the radar itself /5/. Or in other words: the aircraft with an active HiVision radar can be detected earlier by "ears" than by an ESM receiver.

4. Results

4.1 Experimental setup

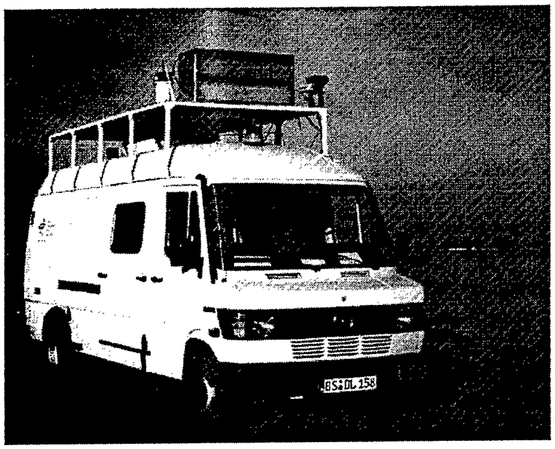


Fig. 4.1 : Multi sensor test van, equipped with the mmw-radar (aluminium box on the top), a video- and an IR-sensor.

The DASA HiVision Radar is a rather new development which was first investigated within a static environment/11/. The Institute of Flight Guidance at DLR Braunschweig started to take the first step from static scenario to a dynamic one in 1996. In order to establish a rather quick experience, a Mercedes Benz van (Fig. 4.1.) was equipped with a differential GPS receiver, with a Litton inertial reference unit (to determine the exact trajectory during the test) and a set of forward looking imaging sensors for the visible and the infrared channel. The visible channel was covered with a video camera type FA 871 (0.4 – 1.0 μm wavelength) and a resolution of 581 x 756 pixel manufactured by Grundig. For the infrared channel the AEGAIS – camera manufactured by AIM Heilbronn (former AEG) for the 3-5 μm wavelength and a spatial resolution of 256 x 256 pixel was used. A set of typical images are shown in Fig. 4.3.

In parallel DASA and DLR developed an airborne version of the HiVision Radar for the DO 228 (Fig. 4.2) . First flight tests are planned for the autumn of 1998.

To avoid any misunderstanding: the "AWACS – like" position of the radar on top of the plane (Fig. 4.2) was only chosen to avoid any kind of disturbance between the already existing IR-sensor at the nose of the plane and the radar. With a width of the radar antenna of 70 - 80 cm it should be always possible to find a position inside the radom, - at least at medium sized or large transport aircrafts.

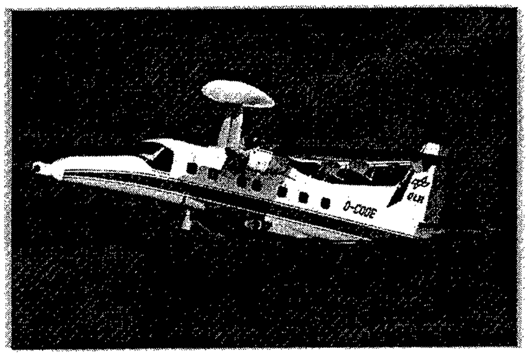
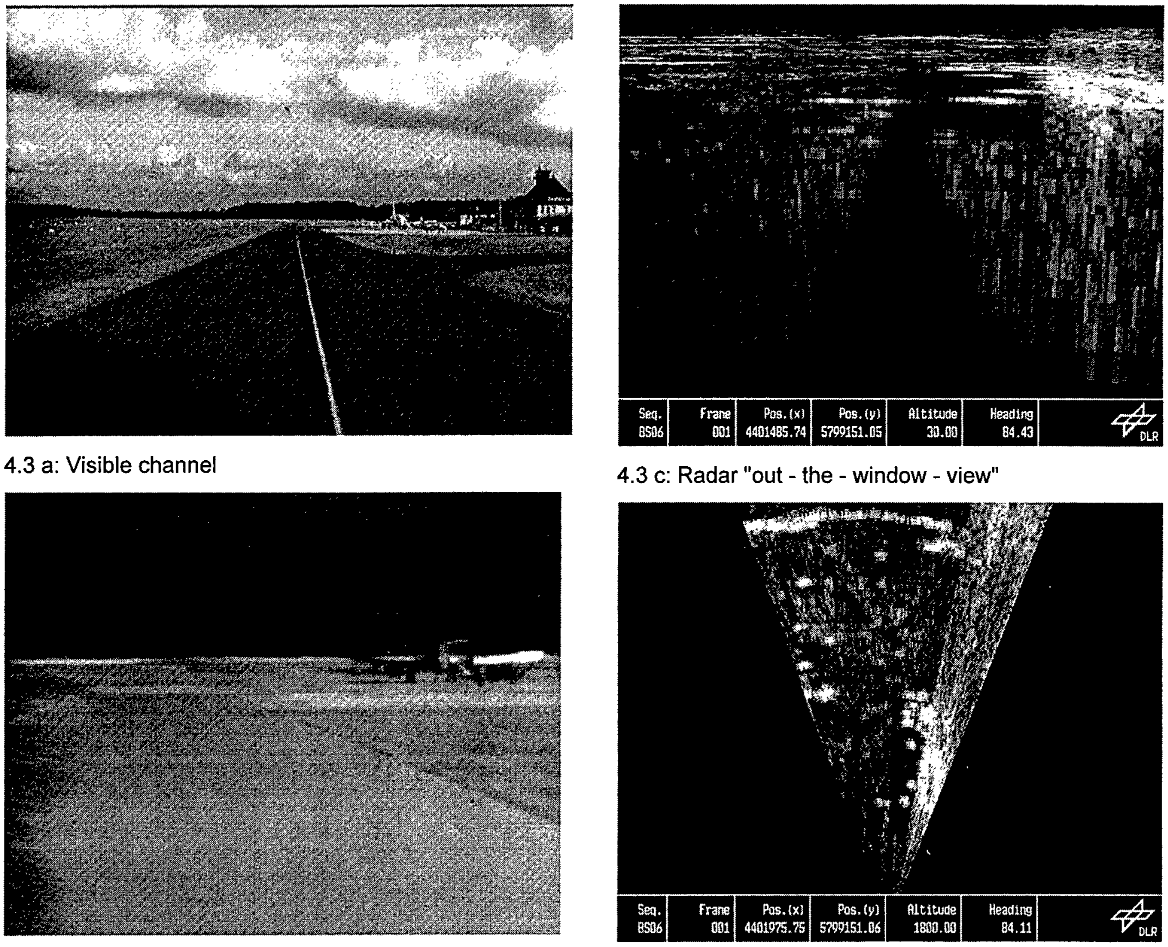


Fig. 4.2 : DASA HiVision radar mounted on top of a DO 228.

4.2 Experiments with DASA HiVision Radar

Sensor data sets were recorded on Braunschweig airfield driving a well defined course along taxi- and runways. The experiments were repeated at different times of the day (even at night) and under different weather conditions , such as sunny and cloudy sky, rain and fog. Furthermore some artificial obstacles like other vans were placed on the track and natural obstacles (Fig. 4.4) appeared by random.

Fig. 4.3 shows a set of images acquired during a typical test run looking along a concrete runway towards an area with parking airplanes. While Fig. 4.3 a) and b) show unprocessed images of the visible and the IR - channel, images c) and d) show a radar-out-the-window-view (C-scope) and a radar-map-view (PPI-scope) calculated from the raw radar data, which are available in a range angle (B-scope) format.



4.3 a: Visible channel

4.3 c: Radar "out - the - window - view"

4.3 b: IR - channel

4.3 d: Radar map - view.

Fig. 4.3: A set of typical multispectral images of a Braunschweig airport taxi way.

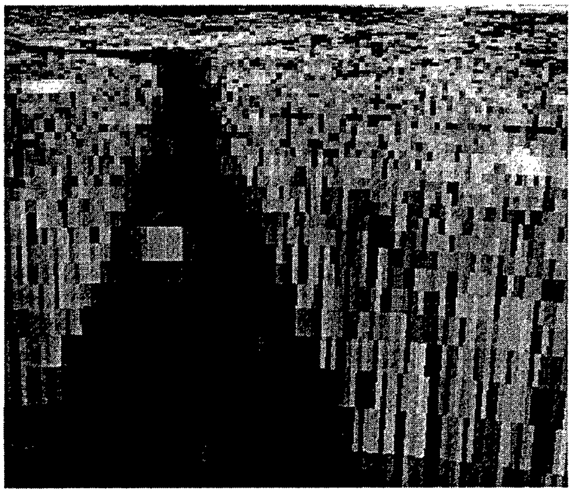


Fig. 4.4: C-scope radar image with "obstacle" on the taxi-way.

A very interesting example of the mmw-radar capability for obstacle detection is presented in Fig. 4.4 This picture shows the same taxi-way on Braunschweig airport as Fig. 4.3c, but this time a certain obstacle was located in the region of the taxi-way. Fig. 4.4 also emphasizes some problems which are connected with the interpretation of a radar image: the poor image resolution and the kind of image generation (range angle) which contains no information about the object's height or its vertical position. The only information which is really transmitted by the radar image of Fig. 4.4 is a reflex in a part of the scene (taxi way) where no reflex would be expected.

The pilot has no idea about the kind of object and due to the way of image generation he has also no information about the object's height and the vertical position (ref. Fig. 4.8). In this special case the pilot would identify a bird sitting on the taxi-way, simply by looking out of the window. That's of course not the idea for an EVS-sensor, but from the pure radar information nobody would really be able to decide whether the "obstacle" is under, ahead of or above the radar position. This problem of radar-image interpretation is still open.

4.3 Data Fusion

An integrated Enhanced Vision System, as it is defined in chapter 2, requires the fusion of several data sources. In order to achieve that, all data has to be transformed into the same level of representation.

Especially a radar image data can exist in different formats:

1.	range-angle	B-scope	Fig. 4.5 a
2.	map	PPI-scope	Fig. 4.5 b
3.	out-the window-view	C - scope	Fig. 4.5 c

Of special interest for radar images is the range-angle, or B-scope representation. This type of picture contains two basic informations: the distance (range) of a certain object relative to the radar head and the direction in terms of the azimuth angle. These "range angle"- pictures use the x-axis for the angle representation and the y-axis for the object (= reflector) distances.

Figure 4.5. shows some examples for these different types of image data representations.

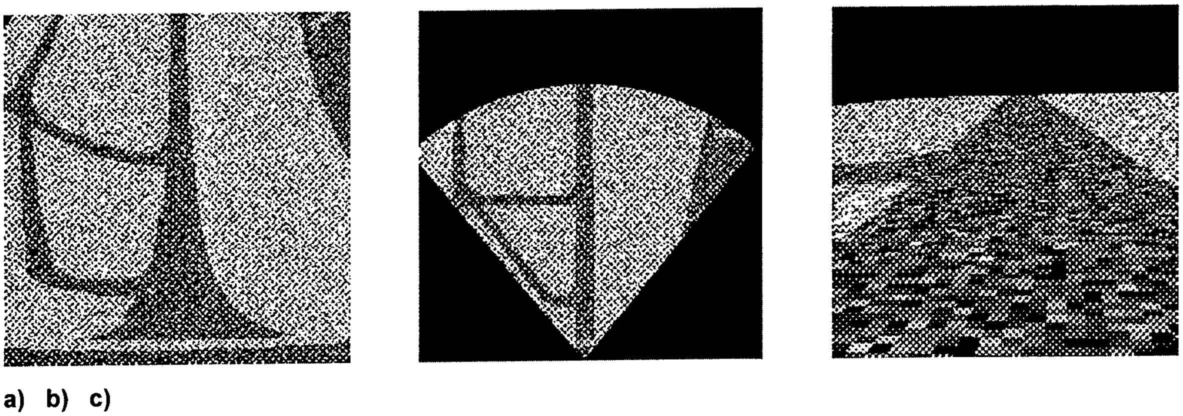
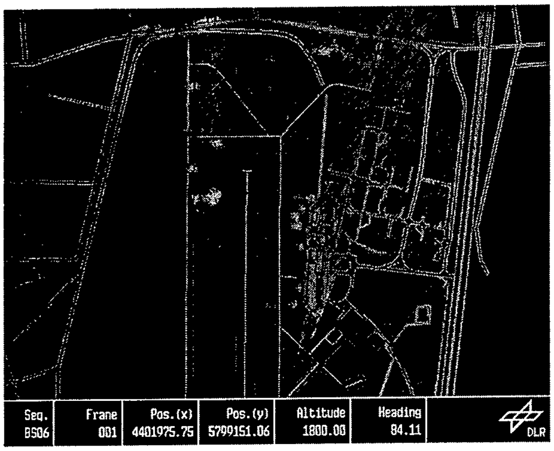
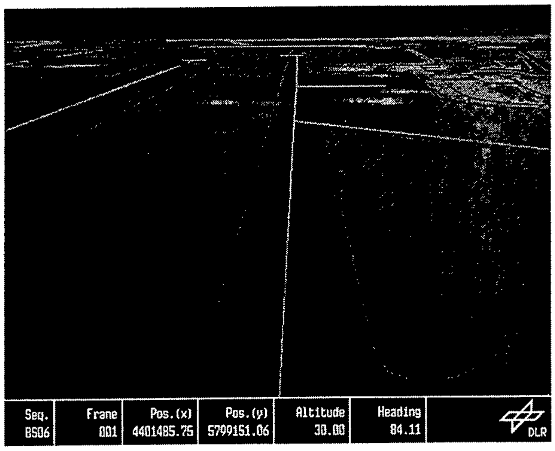


Fig. 4.5: Different radar image representations of the same scene. a. Range-angle (B-scope); b. PPI-scope or range / cross-range; c. "out-the-window-view" (C-scope).



Seq.	Frame	Pos.(x)	Pos.(y)	Altitude	Heading
BS06	001	4401975.75	5799151.06	1800.00	84.11

Fig. 4.6a : Radar image with terrain overlay : Radar-map-view.



Seq.	Frame	Pos.(x)	Pos.(y)	Altitude	Heading
BS06	001	4401485.75	5799151.06	30.00	84.11

Fig. 4.6b : Radar images with terrain overlay : Radar-out-the window-view.

A first, rather simple realization of an Enhanced Vision System deals with the fusion of radar images, which might be available in the "PPI - scope"- format or as "out-the-window-view" with a terrain data base.

Position and attitude delivered by the vehicle's navigation system and a 3-D terrain model are used to compute a wire frame overlay of terrain elements which can be presented in combination with the radar image. Figure 4.6 shows two examples of a radar out-the-window-view (C-scope Fig. 4.6b) and a radar-map-view (PPI-scope, Fig. 4.6a).

The fused radar images are a rather natural supplement to the type of informations which are carried by the primary flight display (PFD). A tentative example how to integrate such a fused image into a PFD is shown in Fig. 4.7.

Fusion techniques, especially in the field of image data fusion, are often devoted to the fusion of several image sources with different spectral sensitivity /13/. In a first step the emission of all objects and radiation sources are transformed into the same spectral region in order to make them visible or detectable with a common tool, - in many cases the human eye.

Disregarding the possibility of data reduction which might be achieved with an image fusion technique -, the main reason for image data fusion is to improve the situational awareness of the pilot and the observer respectively. In /14/ the authors demonstrate a dramatic increase in reliability of target detection and observer performance if visible and thermal images are fused either grey or colour.

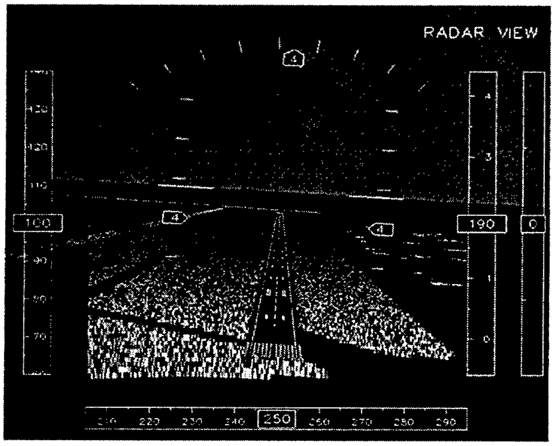


Fig. 4.7: Primary flight display with an integrated fused radar image .

If we concentrate for a moment on the EVS - technique as a possibility to increase the pilot's or crew's situation awareness - again disregarding requirements for reconnaissance or remote control - we have to compete with a very famous "vision system": the pilot's eyes! In terms of situation awareness the EVS - technology should "enhance" the ability of the human operator, but not replace him. With this background the design of an enhanced vision system, especially the applied fusion philosophy, has to be examined very carefully :

- What kind of additional sensor really enhances the human cognition ?
- How useful is a sensor which covers the same, or a similar spectral range as the human eye ?
- What kind of representation (map or "out-the-window-view") in what situation should be used ?
- What type of images , symbols, synthetic- or sensor vision, should be displayed on a HUD ?

Another rather important type of data fusion, especially for radar images, is related to a technique which could be called "data fusion in time and space". Beside the fusion of different data (images) from different sources, it is also possible to fuse different images from one sensor, taken from different positions and at different times . The SAR - technique or computer tomography are good examples how to improve the information of one sensor using data "fusion in space". For EVS applications this approach offers the following advantages:

- (determination of objects height and vertical position by displacement vectors;)
- reduction of background noise (fusion in time);
- integration of single snapshots to an overall view. (Fusion in space)

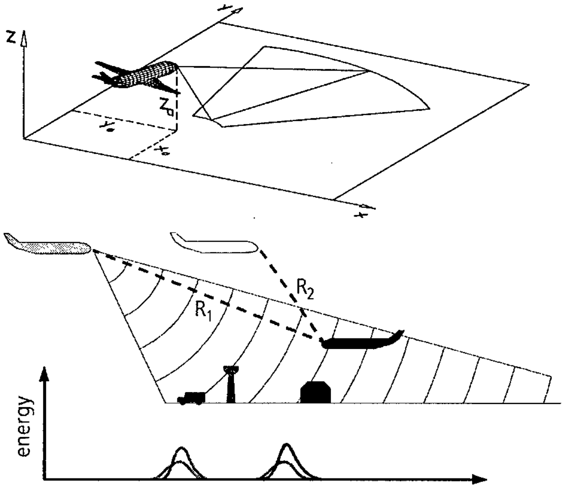


Fig. 4.8: The radar imaging situation:

The first point is set into brackets, because the authors feel that this advantage doesn't really exist, - although it is mentioned in the literature /12/. Figure 4.8 shows the situation. The radar is only able to measure the distance to certain objects, an information about the vertical position is not available. A possible solution of this problem could be a series of distance measurements which are taken from the moving radar, because the slant distance is a function of the objects vertical position relative to the radar head. This is the idea to overcome the lack information about the vertical axis, - but for the actual available radar systems this procedure fails, due to the rather large range errors, which are in the order 3 – 6 m.

radar height (m):	200	300	400
distance between radar and the object for the 2. measurement :	1)	1)	1)
R_2 = 900 m	1)	1)	260m 570m
R_2 = 800 m	1)	1)	283 m 483 m
R_2 = 700 m	1)	201 m 370 m	2)
R_2 = 600 m	1)	243 m 346 m	2)
R_2 = 500 m	133 m 248 m	2)	2)

Table 4.1 : Errors of the vertical position measured by a moving radar. (Starting distance between radar and object: 1000m; range error of the radar: 6 m) 1) : No existing numerical solution. 2) : Object not within the elevation range of the radar.

The following example from table 4.1 clarifies the basic procedure: the vertical distance between the radar head and the object might be 400m; the slant distance R_1 to the object is assumed to be 1000m. In order to get an information about the vertical position of the object a second measurement at a slant distance of R_2 = 900 m is carried out. Both distance measurements are afflicted with a range error of 6m, which leads to an uncertainty of the vertical position between 260m and 570 m.

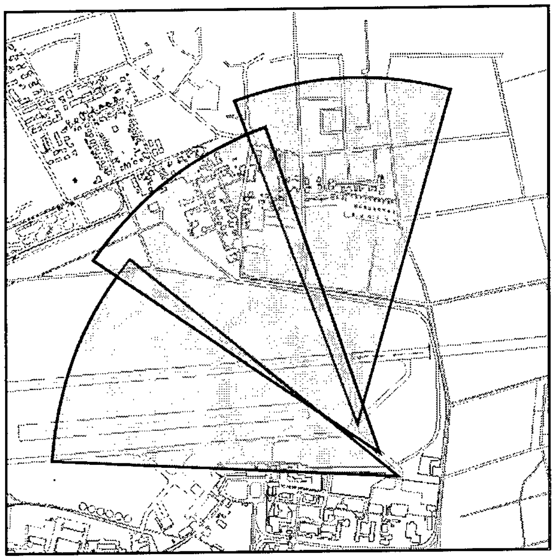


Fig. 4.9: Fusion of radar images to an overall view of Braunschweig airfield. (Fusion in space, with a moving vehicle)

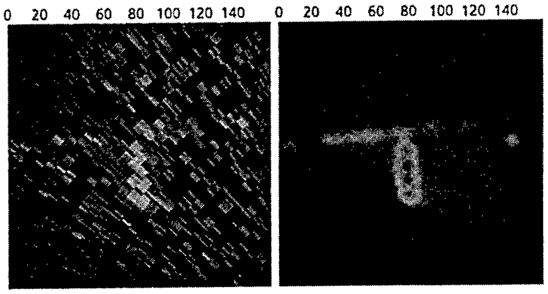
The other above mentioned data fusion techniques - fusion in time and fusion in space - are rather promising, particularly for the range angle images of a radar. Fig 4.9 shows some details of a fusion in space: The wire frame map in Fig. 4.9 represents the Braunschweig airfield and the shaded segments indicate those parts of the map which are covered by the radar (PPI-scope) during taxiing. Using the information about the vehicles position, which is given by an inertial reference unit, or by means of GPS, every single radar image can be assembled to an overall view of the whole airfield. (For this special fusion technique the range-angle-information of a radar picture is very advantageous.)

The image rate of the DASA HiVision radar is 16 Hz, which leads to a large overlap of the PPI-scope pictures. If the vehicle's velocity is rather slow compared with the frame rate, - which is a typical situation during taxiing -, the overlap between consecutive radar images covers a large common area. This feature can be used for noise reduction and to eliminate or to detect moving targets by means of an averaging technique, adding up weighted grey scale values. The effect of such a motion compensated fusion technique is threefold:

- noise reduction;
- enlargement of the image coverage;
- increase of spatial resolution.

The result is shown in Fig. 4.10: Fig 4.10a presents a detail of a raw PPI-scope picture. After fusion in space and time, the picture noise was nearly completely eliminated and the object itself can be identified as a row of lamps.

The fusion of radar images shown in Figure 4.9 refers to a situation with a fixed map and moving vehicle. In this case the fusion procedure eliminates all moving objects and of course the noise. These features qualifies this type of fusion as a "map generator".



a.) b)

Fig 4.10: Noise reduction in a PPI-scope image . a: detail of single radar image. b: the same part of the picture after fusion.

Reference System :	Fused type of image :	Result:
Cockpit	C-scope	noise reduced "out-the-window-view"
Geographic system	PPI-scope	overall map - view

Table 4.2: Fusion in time and space for different reference systems.

The same fusion technique can be applied to a "out-the-window-view" (C-scope) with a fixed vehicle and a moving map. Unfortunately moving - and comoving objects are also suppressed, but the visibility of fixed objects will increase. Table 4.2 compares these different fusion techniques.

5. Conclusion

The integrated enhanced vision concept as it is investigated at DLR demands the fusion of sensor vision, synthetic vision and additional informations of the inner and outer status of the vehicle. In order to meet most weather situations DLR decided to use the HiVision radar from DASA Ulm, which is a forward looking active , - but from the military standpoint of view a 'silent' -, mm-wave imaging sensor.

Radar sensors are very promising in the field of weather penetration, but on the other side they are a great challenge concerning noise and resolution, (angle and range) and they fail completely delivering the vertical position of the depicted object. Data fusion techniques, such as an overlay with terrain data, or fusion in time and space may be applied to overcome the former difficulties. The restriction of the mmw-technology to a range - angle information and the impact on the architecture of the overall Enhanced Vision System is still a topic for research.

A very promising imaging technique in this field would be the total 3-D reconstruction of the outside world from 2-D (range-angle) radar images. This would require the adoption of signal reconstruction methods, as they are known from computer tomography by fusing radar images with different scanning directions. These image data could be acquired for example by at least two orthogonal mounted radar antennas or by a rotating one.

First experiments with the DASA HiVision Radar as an enhanced vision sensor show some very promising results concerning range, image rate and resolution.

6. References

/1/ Hester, R.B.; Summers, L.G.; Todd, J.R. "Seeing Through the Weather: Enhanced / Synthetic Vision Systems for Commercial Transports." SAE International Technical Paper Series #921973, Aerotech '92, 5-8.Oct 1992; p 97 - 104

/2/ "Enhanced and Synthetic Vision 1997", Jaques G. Verly, Editor, Proceedings of SPIE Vol. 3088, 1997

/3/ Horne, W.F.; Ekiert, S.; Radke, J.; Tucker, R.R.; Hannan, C.H.; Zak, J.A., "Synthetic Vision System Technology Demonstration Methodology and Results to Date", SAE International Technical Paper Series #921971, Aerotech '92, 5-8 Oct 1992; p. 1-19

/4/ Nordwall, B.D.; "UV Sensor Proposed As Pilot Landing Aid", Aviation Week & Space Technology, August 11, 1997, p. 81-84.

/5/ Pirkl, M.; Tospann, F.J.; "The HiVision MM-Wave Rasdar for Enhanced Vision Systems in Civil and Military Transport Aircraft." SPIE, Vol. 3088, "Enhanced and Synthetic Vision 1997" p. 8-18 (1997)

/6/ Shoucri, M.; Dow, G.S.; Hauss, B.; Lee, P.; Yujiri, L.; " Passive milimeter-wave camera for vehicle guidance in low-visibility conditions." SPIE, Vol. 2463, p. 2-9 (1995)

/7/ Appleby, R. ; Price, S.; Gleed, D.G.;
"Passive milimeter-wave imaging: seeing in very poor visibility"
SPIE, Vol. 2463, p. 10 – 19 (1995)

/8/ Shoucri, M.; Dow, Fornaca, S.; G.S.; Hauss, B.; Yujiri, L.; " Passive milimeter-wave camera for enhanced vision systems." in Enhanced and Synthetic Vision 1996, J.G.Verly, Editor, Proc. of SPIE, Vol. 2736, p. 2-8 (1996)

/9/ Doehler, H.-U.; Bollmeyer, D.; "Simulation of imaging radar for obstacle avoidance and enhanced vision." in "Enhanced and Synthetic Vision 1997, J.G. Verly, Editor, Proc. of SPIE, Vol. 3088, p. 64 – 73 (1997)

/10/ Bui,L.; Franklin, M.; Taylor Chr.; Neilson, G.; "Autonomous Landing Guidance System Validation" in Enhanced and Synthetic Vision 1997, J. G. Verly, Editor, Proc. of SPIE Vol. 3088, p. 19 - 25 (1997)

/11/ Tospann, F.-H.; Pirkl, M.; Grüner, W.; "Multifunction 35 GHz FMCW radar with frequency scanning antenna for synthetic vision application2." in Synthetic Vision for Vehicle Guidance and Control, J. G. Verly, Editor, Proc. of SPIE Vol. 2463, p. 28 – 37 (1995)

/12/ Pavel, M,; Sharma, R.K.; "Fusion of radar images rectification without the flat earth assumption.", in Enhanced and Synthetic Vision 1996, J.G.Verly, Editor, Proc. of SPIE, Vol. 2736, p. 108-118 (1996)

/13/ Sweet, B.T.; Tiana, C.; "Image processing and fusion for landing guidance", in Enhanced and Synthetic Vision 1996, J.G.Verly, Editor, Proc. of SPIE, Vol. 2736, p. 84-95 (1996)

/14/ Toet, A.; IJspeert, J.K.; Waxman, A.M.; Aguilar, M.; "Fusion of visible and thermal imagery improves situational awareness", in Enhanced and Synthetic Vision 1997, J. G. Verly, Editor, Proc. of SPIE Vol. 3088, p. 177 - 188 (1997)

Software Testbed for Sensor Fusion Using Fuzzy Logic

Stephen C. Stubberud
Kristi A. Lugo
ORINCON Corporation
9363 Towne Centre Drive
San Diego, CA 92121 USA

SUMMARY

While today's sensors are increasing in accuracy, they are also required to operate in conditions where distortions, clutter, and saturations can have an adverse effect on their performance. Usually, to account for such contaminated sensor data, the covariance of the white noise error is increased. Unfortunately, this approach to account for this external error and noise sources defeats the improved sensor accuracies. Recently, various algorithms have been developed using fuzzy logic and neural network-based techniques to improve the error modeling and processing in multisensor data fusion. The development of a "cookbook" of sensor modeling methods, data association, and data fusion algorithms provides a firm foundation from which to perform a comprehensive study of these algorithms' effectiveness. Recently, we have begun the development of an open architecture software structure that will provide the capability to develop various data fusion implementations by mixing and matching algorithms. This software testbed permits the selection of various algorithms and parameters that provide the user with a capability to analyze the performance of the algorithms in various combinations to determine the most effective data fusion algorithms for specific conditions.

INTRODUCTION

Target localization and classification are of key importance in improving the survivability and lethality of weapon platforms and in reducing the incidents of destruction of friendly troops and assets. For this improvement to be realized, multisensor data fusion, the development a coherent scene from information provide by a myriad of sensors, must be enhanced from its current technological state. To achieve this purpose, the authors are developing a data fusion software testbed to evaluate various data fusion algorithms, that incorporate fuzzy logic, rule bases, and/or neural networks.

Standard data fusion techniques usually rely on a functional flow as depicted in Figure 1. Sensors report information. Depending on the type of filter, i.e., $\alpha - \beta$ filter or a Kalman filter, this information is some measurement or a measurement with an error associated with it. Usually, the error is based on some statistical model, often a Gaussian. This measurement is then fed to the association algorithm, which compares the measurement to the localizations of all existing targets, also known as tracks. If the measurement is deemed close enough to a particular track, then that measurement is filtered into, or fused, with that track. Some techniques such as multihypothesis trackers[1] and n-p completeness-based algorithms[2] have been developed to improve statistical performance of the data fusion systems. These types of algorithms permit the collection of more measurements before making hard decisions about the target localization scene.

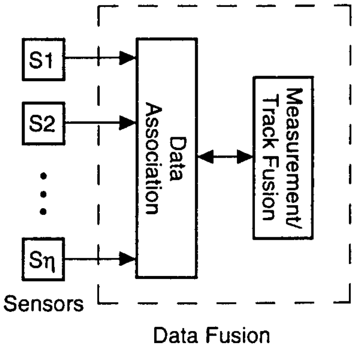


Figure 1. Standard Data Fusion

To incorporate intelligent and adaptive techniques such as those used for fuzzy logic and neural networks, we need to change the intercommunication between the various section of the multisensor data fusion problem. As detailed in Figure 2, all elements of the data fusion system must be able to share information with each of the other elements. From this communication scheme, we designed our data fusion system software testbed. The basic testbed is an open architecture system that will permit more than just various algorithms—it will permit variations within the algorithms, such as changes in the shapes of the various membership functions.

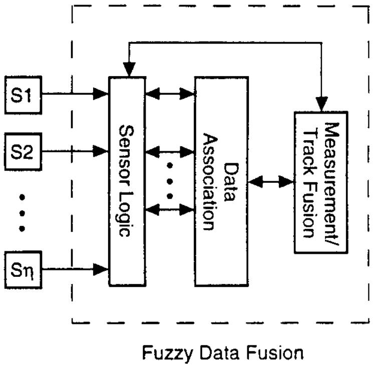


Figure 2. Modern Data Fusion

In this paper, we will discuss the design of the data fusion testbed and provide an overview of some of the proposed algorithms to be evaluated in this system. In Section 2, the overall system design will be presented, including an introduction to so-called environmental maps. Section 3 discusses the algorithms that will be applied to the sensor systems and reports. The data association routines will be discussed in Section 4. In this section, we will describe in more detail the development of a fuzzy logic-based association routine. Section 5 will introduce the various filter, or fusion, routines that will be evaluated in this testbed.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

OVERALL SYSTEM DESIGN

The top-level design of the system, including its main constituent components, is similar to standard kinematic and classification data fusion techniques. The measurements are reported by the sensors. This information is processed and passed on to the association routines, which pair measurements with tracks. These pairs are then fused together.

In Figure 3, we provide a general description of the design of the new fusion system. The changes in this fusion system consist of both the internal communications and the required external inputs. The component of the fuzzy fusion testbed that is different from most standard fusion systems is the environmental maps. Many of the algorithms we have investigated and developed are dependent on spatial information. Thus, the information from the environmental map database is critical. Another important input to the system is the sensor information, which includes static parameters about the sensors, such as beam width and detection ranges, and dynamic information such as signal power.

The intercommunication between the internal processes is also an important element of the new fusion architecture. While the intercommunication is essential, it also causes the greatest risks. The intercommunication, if not handled properly, can result in the development of a feedback loop, which would behave similar to a control system positive feedback loop. This situation requires that we take special care with each algorithm to maintain the integrity of the processed information.

SENSOR PROCESSING

Figure 4 breaks down the sensor logic component of the fusion system into its main subcomponents. These subcomponents group the various algorithms by both their input requirements and their resulting outputs.

As noted, there are five current components to the sensor logic system, including two sets of error models, one that requires environmental maps and one that does not. Two of the components have algorithms that process the measurement and its error covariance. Finally, the last component is the set of algorithms that fall outside of the realm of any other subcomponent.

From a software design point of view, the classification of components based on the similarities among the inputs and outputs of the various algorithms is quite natural. However, in the technical aspects of data fusion, sensor logic routines can broken down by the classification of the condition or variable that has primary effect on the algorithm. The following is an extensive list of conditions:

Range
Range Differential
Azimuth Differential
Variable Environment Conditions
Stationary Environment Conditions
Sensor Registration
Sensor Peculiarities

Changes in the behavior of sensors and their measurements can also result from overlapping areas. For example, one of our test algorithms is from the software component "other." This algorithm is based on angular resolution (beam width of the sensor) and the range to the target. As shown in Figure 5, two targets can be indiscernible at long ranges while they will become distinct at shorter ranges. By using the measurement and the static sensor information about resolution, we can calculate a probability that two targets are approaching. This information can be later used to flag the single target track as a potential multiple target. This technique for track monitoring breaks with the tradition that a single measurement is interpreted as a report from a single target.

More complex sensor algorithms have also been developed, including techniques to incorporate spatial information such as the existence of agricultural sites to discern between false alarms and targets. A simple example is wind blowing across leafy vegetation. Radar returns will contain not only locations, but also Doppler shifts. Thus, we have a moving target. In agricultural settings, the near-uniform spacing can cause greater problems in that it can create clutter that emulates tracks with straight-line motion. For routines that handle such problems, required inputs are stationary environmental maps, which pinpoint the locations of fields, local wind speeds, measurements, and locations of existing target tracks. Such information is folded into the inference engines to estimate the probability of the existence of background clutter and the ability to correct measurements and adjust covariances accordingly.

All the algorithms that have been developed for this software testbed can only require, from external sources, measurements, raw values such as percentages of return power from a radar pulse, sensor parameters such as antenna frequencies or sensor array length, and known spatial information. All internal values such as classifications, track localizations, and association scores will also be available as inputs. Finally, as part of the open architecture design of this system, all neural network and fuzzy logic routines will be based such that they can use any of the library of squashing functions and membership functions that are available.

The software design, developed from the interpretation of error sources, permits a wider range of sensor logic development and testing than is currently slated for implementation.

DATA ASSOCIATION

Data association is the comparison of measurements to existing tracks to determine if a new measurement is being reported from an existing target. The localization of the measurement and the predicted localization of a track are compared, and the distance between the two is computed. This distance may or may not be normalized. If the distance is within a given threshold, then the measurement is said to associate with that track, and a score is calculated. A measurement can associate with more than one track. The scores are used by various techniques to handle such conflicts.

Most modern sensor data fusion systems use the residual chi-squared metric

$$\chi^2 = (z - \hat{z})^T (HPH^T + R)^{-1} (z - \hat{z}) .$$

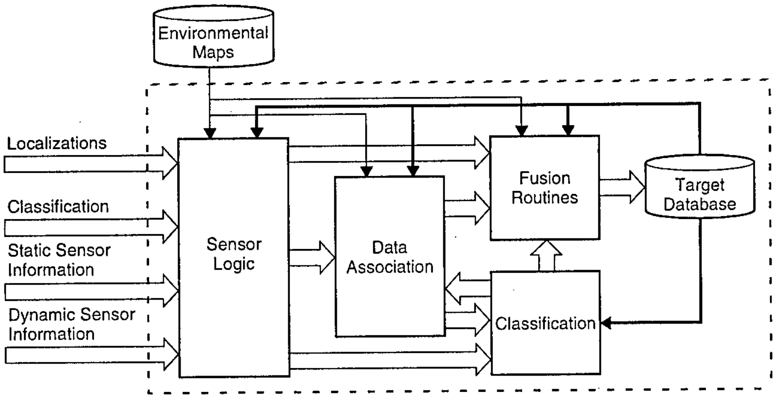


Figure 3. Fuzzy Fusion Software Testbed Design

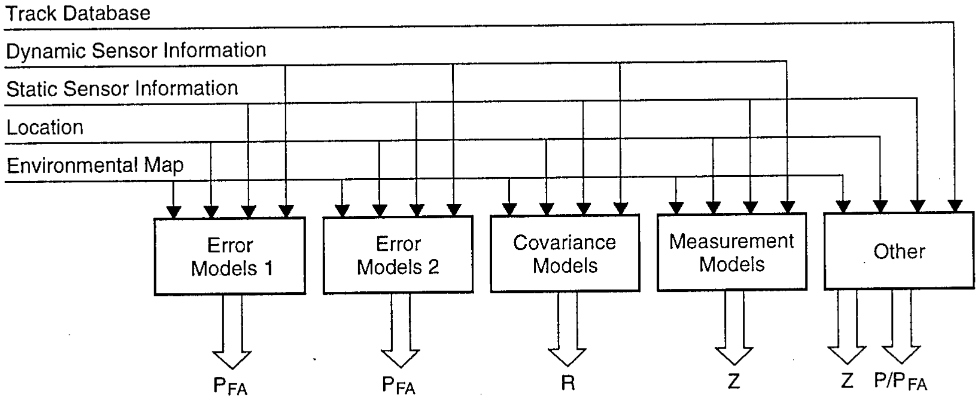


Figure 4. Sensor Logic Implementation Design

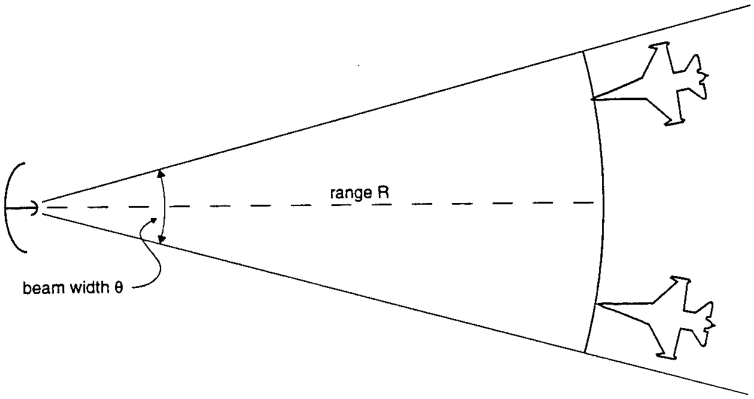


Figure 5. Two Targets Seen as One Because of Angular Resolution

The term z is the measurement, while $\hat{z}$ represents the predicted measurement of an existing track from previous measurement information. The terms R and P are the error covariances of the measurement and the predicted track, respectively. The matrix H is used to convert the error covariance of the track in the same coordinate system as the measurement.

For this fusion testbed system, this association technique is referred to as standard association. So far, for this system, we have planned to implement four association routines that, as shown in Figure 6, are partitioned into three separate components. These components include the standard association discussed above, fuzzy association techniques, and a post association module. The fuzzy association algorithms consist of a fuzzy logic version of Joint Probability Data Association (JPDA), which is described in [3] and [4], and a fuzzy logic-based association technique. The post association algorithm permits the incorporation of probability of false alarm information in the measurement noise covariance.

Both the standard association technique and JPDA require that all measurement noise covariances are Gaussian. However, these covariances are not always Gaussian, which is why the development of a fuzzy association technique was considered. As shown in Figure 7, the standard association equation can be interpreted as the amount of probability overlap of an n-sigma ellipse based on the statistics of the measurement and n-sigma ellipse based on the statistics of the target track. However, as noted before, not all error and noise statistics are Gaussian. For example, if we incorporate spatial information such as sensor blockage, the noise and error ellipses can change radically (Figure 8).

The concept of the fuzzy data fusion algorithm is to determine the region of overlap and to integrate that information into the association score. After removing the error ellipse portions that violate spatial conditions, we compute the residual between the measurement and the predicted measurement, as with the standard association. Second, we determine the covariance sizes. The membership functions depicted in Figure 9 are used.

Using an inference engine, we set the parameters of the membership functions that define the residual size, as depicted in Figure 10. Depending on which membership function fires, the inference engine that computes the association score based on the percentage of covariance overlap of the track and the measurement is fired. As shown in Figure 11, each inference engine has one variation in input and this is the residual difference.

In summary, we incorporate spatial rules into the error ellipses to remove areas that are violation. We compute the residual and determine its linguistic size relative to the error covariances. Depending on the residual size, inferences that relate to the percentages of elliptical overlap are fired. The resulting association scores are normalized.

We note that we do not depend on Gaussian properties at all, implying that non-Gaussian covariances are permissible.

DATA FUSION

The final major subcomponent of the sensor fusion testbed is the actual fusion algorithms. As depicted in Figure 12, the fusion component is broken into three sets of algorithms: standard, augmented, and intelligent.

The standard fusion algorithm is the Extended Kalman Filter (EKF). The EKF requires the track state (predicted or updated), its error covariance, the measurement, and the measurement noise covariance. The augmented fusion algorithms are a reduced-order Kalman filter, similar to that described in [5], augmented with a neural network to filter the measurement noise, and a boundary condition tracker that works in conjunction with the standard EKF, the environmental maps, and classification to prevent target tracks from entering forbidden terrain.[6]

The intelligent fusion routines are the algorithms that require the most design concern. The algorithms include a total fuzzy Kalman filter,[7] an EKF that can process fuzzy measurements,[8] an EKF whose motion model is augmented with an adaptive neural network,[9,10] an EKF whose state is corrected by a neural network,[11] and an EKF whose model is provided by an adaptive fuzzy logic routine. While we will not discuss the details of these algorithms in this paper, we will discuss their implementation details.

The fuzzy Kalman filter requires the most attention to detail. All of the states, measurements, and covariances can be fuzzy membership functions. While the reports are crisp numbers, the stored values are not. Each "value" becomes an array that describes the membership functions that comprise the final membership function. This means that, for each element of the state and covariance matrix, we need to know how many membership functions are involved, the degree of membership of each function, and the shape of each function. So we end up with vectors and matrices of arrays.

The two other techniques that require comparable implementation concerns are the neural network-based methods. Each method requires training, using an EKF-based trainer. This trainer is integrated into the state estimator EKF. Thus, the computation costs increase greatly because of the large number of weights that can be required. To alleviate this computation bottleneck, we exploit both the symmetry and the scarcity of the portions of the matrices that are associated with training the weights.

CONCLUSIONS AND RECOMMENDATIONS

The fusion testbed and algorithms discussed in this paper have been developed and implemented in an effort to determine the usefulness of neural networks and fuzzy logic in sensor data fusion. Recently, papers have begun to appear in the literature that indicate that these new technologies are being considered seriously to improve sensor fusion algorithms. The purpose of this software testbed is to provide a system to the US Army to test such algorithms and to measure their effectiveness. Only by comparison to current and developing algorithms can these new technologies be adequately tested and their effectiveness demonstrated.

The testbed is also designed to be modular to provide a software architecture that may one day go beyond a testbed to a system that can implement several algorithms that may be chosen using automated techniques. This design stems from the philosophy that no one way is optimal for all situations.

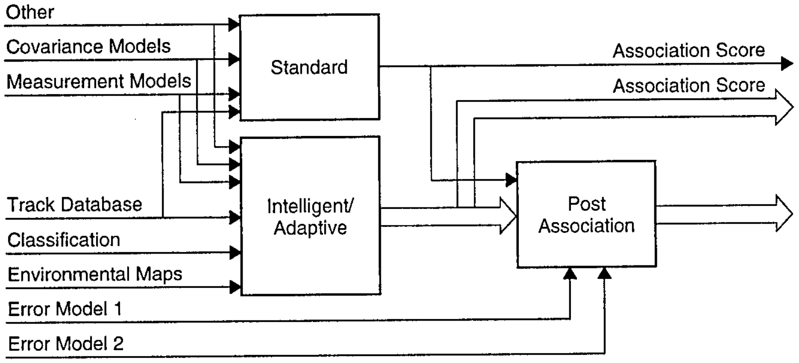


Figure 6. Data Association Implementation Design

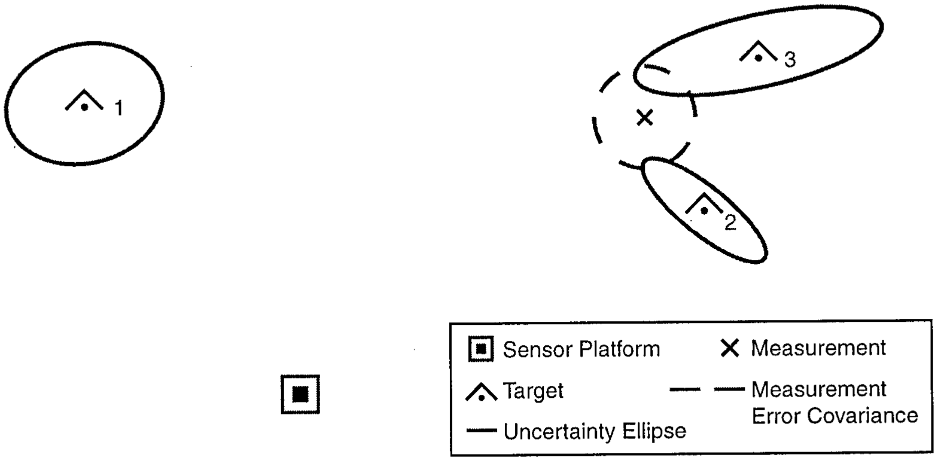


Figure 7. Association of Measurements to Tracks

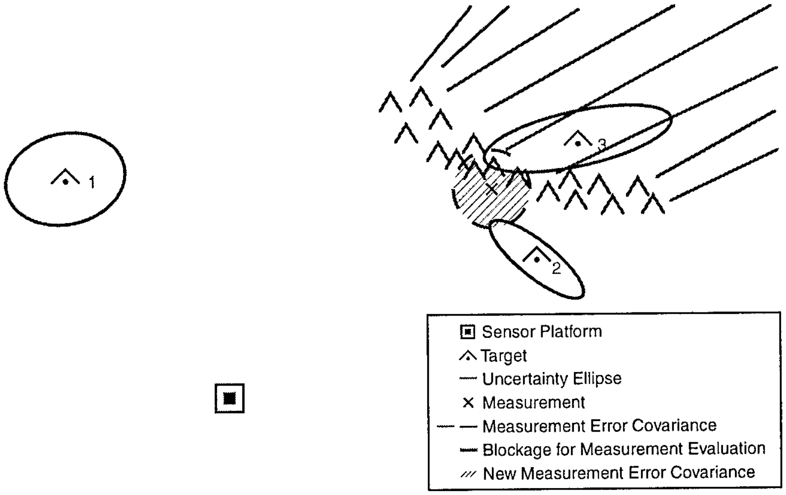

1
3
2
Sensor Platform
Target
Uncertainty Ellipse
Measurement
Measurement Error Covariance
Blockage for Measurement Evaluation
New Measurement Error Covariance

Figure 8. Blockage Affects Sensor Reports

The potential of soft-computing methods for mission systems: A tutorial

A.J. van der Wal
TNO-FEL Physics and Electronics Laboratory
P.O. Box 96864
NL-2509 JG The Hague
The Netherlands
e-mail: vanderwal@fel.tno.nl

1. SUMMARY

In the past decade information processing in general and signal processing in particular has undergone a revolutionary shift from pure AI-oriented methods towards a wide diversity of nonlinear, soft-computing methods that often have a paradigm in biological systems. Among the chief characteristics of these methods are their ease of application, synergy through nonlinearity, their robustness, human-friendliness, the ability to handle ambiguous (even conflicting) information, the intrinsic ability to handle vague notions and do human-like inferencing, the possibility to take into account multiple goals, their learning capability, and the separation of concerns. Although soft computing has proven to be successful in a growing number of application areas, a general theory encompassing all soft-computing methodologies together with standard linear processing methods is still lacking. In soft-computing literature it is seldom discussed *why* a particular method has been selected, or *why* a particular approach is advantageous for solving a problem. In this paper we present a concise discussion of the characteristics of the key softcomputing technologies and their possible impact on the design of mission systems. Because mission systems show a trend towards higher levels of autonomy, the complexity of these systems will increase significantly. At the same time there exists a tendency towards miniaturization, *e.g.* to avoid detection (UAVs). These two competing developments can only be reconciled by the integration of softcomputing methods into mission systems.

2. INTRODUCTION

2.1 Mission Systems

A mission system consists of an ensemble of hardware and software that is aimed at the successful completion of the mission. In this sense many systems may be considered as a mission system, depending on their level of complexity and autonomy. The concept of a mission is well-established in the aerospace community, although landbased and maritime operations face similar problems on a conceptual system level. Missions are generally initiated by man and still the majority of missions is manned to ensure that unexpected and difficult situations can be dealt with adequately. In addition we observe an increasing interest in *unmanned* missions. This is caused by changes in the world situation that complicate the execution of full-blown missions, but at the same time require mission systems to be extremely well-informed. In this context unmanned, robotic reconnaissance vehicles such as UAVs are developed.

Recent history shows that the nature of military operations changes rapidly: Although sensors are vital to the success of any military mission, it becomes at the same time much more difficult to interpret these observations. This can be illustrated by the introduction of stealth technologies (radar), by which planes become much more difficult to detect by radar, the subtleties of ‘peacekeeping’ missions compared to classical, full scale warfare scenarios, and finally the complexities and greater vulnerability of navy vessels operating close to shore (‘littoral warfare’). Finally it should be noted that there is a genuine need to fuse sensor generated information, at least at the higher levels of command and control: the man-machine interface being the limiting factor. Although new sensors

Paper presented at the RTO SCI Symposium on “The Application of Information Technologies (Computer Science) to Mission Systems”, held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

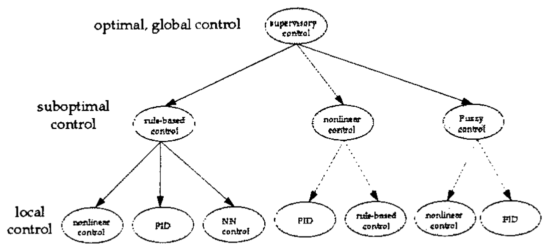


Figure 1 *Hierarchy of autonomous control processes in a mission system.*

have been developed (e.g. GPS) and accuracy and resolution in space and time of most existing sensors have greatly increased in time, the bandwidth of the man-machine interface has not. The situation of having to deal with more information than one can process in a certain time is not unsimilar to the situation where a *lack* of information exists. Both situations involve taking decisions in the presence of uncertainty and would benefit from intelligent data reduction techniques, such as softcomputing.

Other reasons for initiating unmanned missions are the safety of personnel and new tactical doctrines. Even in unmanned missions there may be on-line control by man via *teleoperatio*n, resulting in *telepresence*, but the mission can also be fully autonomous. Simulation may play an increasingly important part in the development and testing of a mission system prior to its deployment. Another important issue in mission systems (either manned or unmanned) is the man-machine interface (MMI), despite of (or sometimes because of) the ever-increasing speed and complexity of computer systems. Although it is not easy to give a definition of a mission system, we will in the present paper use the following working definition: *"A mission system is a system that supports the goal of a mission at a certain level of autonomy by optimizing subtasks".*

In this recursive definition subtasks may equally well be viewed as mission systems themselves and thus a hierarchy of different mission tasks is defined at a number of levels of autonomy (Fig.1). A specialized control system has a particular task with only limited autonomy on the mission system level, whereas a software package for overall mission management will have more autonomy. Yet both systems are mission systems. Typically a mission management system will be responsible for high level goals such as the allocation of resources, the scheduling of tasks at certain phases in the mission, as well as providing adequate information to the humans responsible for the mission (Fig. 2). In fact the key issue in mission management is *optimization.* An example is the allocation of resources, *e.g.* fuel, computing power, the supply of energy, information, sensors, *etc.* But also subsystems such as controllers aim to optimize their functioning, *e.g.* by selftuning , and robust performance. The goals of subsystems will often be competing with each other. The central problem to be studied in relation to mission systems therefore is that of *global* optimization *vs. local* optimization.

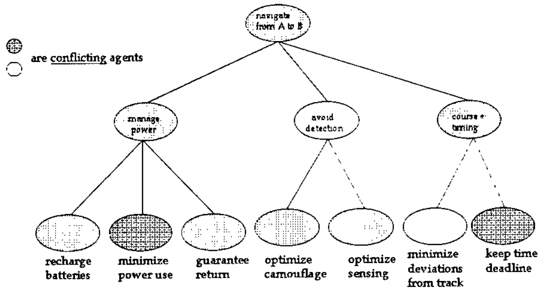


Figure 2 *The mission "Navigate from A to B" consists of many (sub)missions. Completing a mission successfully involves global optimization, which is hard because many competing processes take part in the process. The dark and white-colored processes indicate competing agents.*

2.2 Softcomputing

In the past decade information processing in general and signal processing in particular has undergone a revolutionary shift from pure AI-oriented methods towards a wide diversity of nonlinear, soft-computing methods that often have a paradigm in biological systems. Among the chief characteristics of these methods are their ease of application, synergy through nonlinearity, their robustness, human-friendliness, the ability to handle ambiguous (even conflicting) information, the intrinsic ability to handle vague notions and do human-like inferencing, the possibility to take into account multiple goals, their learning capability, and the separation of concerns. Although soft computing has proven to be successful in a growing number of application areas, a general theory encompassing all soft-computing methodologies together with standard linear processing methods is still lacking. In soft-computing literature it is seldom discussed

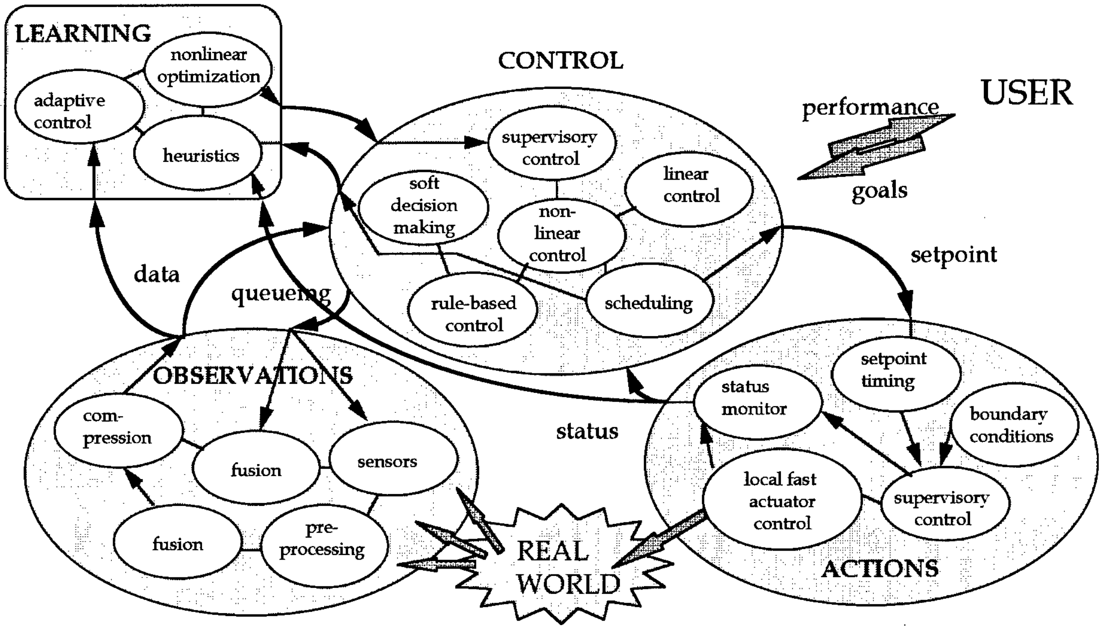


Figure 3 *The fundamental processes within a mission.*

why a particular method has been selected, or *why* a particular structure is advantageous for solving a problem.

Even more rare is a comparison between various possible approaches, *e.g.* selecting between the various possible neural architectures. Thus a novice to the field is confronted with the question how to *select* a suitable method and how to *combine* methods with each other and with more classical approaches. In this tutorial we will discuss the basic theory and properties of fuzzy logic systems (FS), neural networks (NN), genetic algorithms (GA), and the concepts of ordinal optimization (OO), and how these methods may be combined in solving problems in practical applications. A typical example of such an application is *sensor fusion*. Although in this case the uncertainty modeling is usually handled by probability theory, soft computing methods are becoming increasingly more popular. The advantages of a soft-computing approach in comparison to standard approaches have already been recognized in the field of industrial control. In military missions the modeling of uncertainty is more relevant than in industrial applications. For this reason the use of fuzzy measures in decision theory may be relevant to this field. This can be illustrated in the example of sensor fusion. By attributing to each sensor in the sensor suite a unique weight, a general fuzzy measure can be defined in a self-consistent way. With this measure it is possible to take into account non-exhaustive and non-overlapping hypotheses, and also to reduce the cardinality of the space of alternatives, thereby avoiding the combinatorial explosion that is characteristic for *e.g.* the Dempster-Shafer theory of upper and lower probabilities. All soft-computing methods have in common that they somehow carry out a *nonlinear* optimization. As is well-known from optimization theory, heuristics plays an important role, since in practical situations exhaustive optimization is not feasible because of the NP-completeness of the problem. In contrast, soft optimization generates a *soft*, *i.e.* approximate, optimal solution to such problems in only polynomial time. Depending on the type of problem, this can *e.g.* be done by *ordinal* optimization, which is in practice often sufficient. In the final part of the review we will address the issue of merging these soft-computing methods with each other and in relation to classical, analytical methods, so as to take the best of both worlds. Finally it should be noted that although the implementation of these methods can be done on normal digital computers, softcomputing algorithms are intrinsically parallel and thus may benefit from massively parallel, and distributed processing architectures.

2.3 How mission systems can benefit from softcomputing

Mission systems essentially have the internal structure of a control process. By breaking down the overall mission goal one obtains a collection of interacting subprocesses (*cf.* Figs. 1-2). In Fig. 3 the basic functionality of a mission system is outlined: 1) observe, 2) infer (process, interpret, control) and 3) act. Through this cycle the mission system observes the world and acts on it, which action in turn gives rise to changed observations. Two extra functions are added to the scheme of Fig.3, *viz.* 1) interaction with the user and 2) learning, a process aimed to improve the underlying control loops. The learning process may itself be seen as a mission process at a higher, more abstract autonomy level. There is no opportunity in this tutorial to discuss every aspect of how a mission system may benefit from softcomputing, but as an example we will concentrate on the observation process.

In recent years, numerous research papers have been published dealing with the application of multisensor data fusion, also referred to as distributed sensing high-level fusion, especially in the domain of military observations [1-6].

Historically the idea of sensor fusion is not new: As early as the sixties multi-radar trackers have been in use by the military for air traffic control and air defense. Multisensor data fusion seeks to combine information generated by multiple sensors to achieve goals that would be very hard or impossible to achieve with single sensors. From the point of view of efficiency, scheduling, accuracy, and redundancy it seems intuitively obvious that several sensors are 'better' than a single sensor.

Nowadays data fusion is a well-accepted method for making superior inferences in the field of industrial automation (*e.g.* for controlling a power plant, an oil refinery, a cement kiln (for a review on industrial applications, see *e.g.* [7,8]), or even a nuclear reactor [9,10], and for carrying out real-time pattern recognition in industry using a variety of sensors. Especially since the advent of softcomputing methods, such as fuzzy logic, data fusion has become a widely accepted successful fusion technology in industry. We note however that the success of such methods is primarily due to their ability to model human behavior or expertise in supervisory control. Sensor fusion also endeavors to mimic cognitive processes in humans by absorbing the signals of the human observation system, our five senses, from the real world and integrate, or 'fuse', these signal streams to build a coherent picture of our environment. As such, sensor fusion is concerned with lower abstraction levels, much higher information rates, and generally requires faster response than the data fusion used in supervisory control systems. This forms also the key problem in applying soft computing methods to this field: in controlling complex industrial or organizational processes at relatively long timescales, human operators have accumulated over the years ample experience. In contrast, there is only limited insight in the way a human being builds up an environmental picture, his awareness, from continuous multisensate observations.

Although sensor fusion is important to virtually all phenomenological sciences and engineering disciplines, most work until now has been done in the field of *defense* research. This can be understood as follows. In *analytical* approaches, *e.g.* in a physics experiment, the measured quantities or interactions are often so small that the experimental setup has to be designed in such a way as to make sure that the desired quantity or effect is optimally measurable. If the measured quantities are small, the experiment is repeated many times and ergodicity and statistics are used to arrive at average values with low relative standard deviation. Especially in case one tries to prove or disprove the correctness of a theoretical model, this often is a good approach. A final point to note here is that - apart from intrinsic physical real-time aspects - such experiments generally can be repeated many times and real-time constraints are not a bottleneck.

In engineering approaches the use of sensors is more *synthetic*, as illustrated *e.g.* in the field of factory automation. Here one deals with a well-defined problem such as the quality control of products on a manufacturing line, *e.g.* checking the soldering joints on a PCB with an automated vision system. This problem certainly has real-time aspects, but the optimization can be done off-line and the observation circumstances, like in the physics experiment, can be optimized off-line, *e.g.* by testing the best combination of sensors, the proper cameras and illumination, and parallel operation with more than one quality control station if the speed of production requires so.

In military observations we deal with a situation that is far less comfortable than the situations described above: generally speaking it is necessary to assess in real time an often complex situation, that almost certainly is outside one's complete control. Handling such observations requires the modeling of *uncertainty*. Apart from the ordinary problems such as noise and clutter, radar and electro-optical sensors operate also under adverse weather and atmospheric conditions, without any possibility to improve the circumstances of the experiment, or to repeat the experiment, under strict real time constraints, with sometimes enormous consequences of false classification and even more serious penalties for non-detection. In addition, by the nature of the military métier, most interesting targets move at high speeds, try to avoid detection actively or passively, or mislead sensors by jamming or using decoys, and they are designed in such a way as to present a minimal scattering cross section to commonly used sensors and thus to be virtually invisible ('stealth').

Under such circumstances it is clear that doing military observations invariably implies the modeling of uncertainty. Classically this is often done by applying statistical methods, notably Bayes' theorem to formulate a (multi-) hypothesis testing problem. It is however also clear that statistical uncertainty can only model part of the uncertainty. The different measures of uncertainty are now well established in classical set theory, fuzzy set theory, probability theory, possibility theory and evidence theory [11].
The breakdown distinguishes *fuzziness*, or vagueness due to a lack of definite and sharp conceptual distinctions and *ambiguity,* the situation where we are dealing with one-to-many relationships in the information obtained from sensors, yielding *non-specificity* in the case that the data leaves two or more alternatives unspecified, or even *discord*, *i.e.* disagreement in choosing from among several alternatives.

Recently methods that explicitly deal with ambiguity and partially overlapping hypotheses such as Dempster Shafer theory [12,13] and the application of belief functions instead of probability densities have become popular. Of even more recent date is the application of general fuzzy measures [14]. The difficulty inherent to making accurate observations in military applications and the lack of measurement statistics are the prime motivations to improve single sensor observations by merging (partial) inferences/conclusions from one sensor with inferences from the other one.

3. SOFTCOMPUTING

In this section we will give an overview of the principles and basic concepts of four basic softcomputing techniques, *viz.* fuzzy systems (FS), neural networks (NN), genetic algorithms (GA), and ordinal optimization (OO) from the point of view of their potential use in mission systems. Emphasis is placed on the similarities of these four techniques stressing their ability to model complex nonlinear relationships in a multidimensional world. All these softcomputing methods can be applied in universal function approximation schemes (*e.g.* pattern recognition) and in nonlinear optimization.

Both pattern recognition and optimization (*e.g.* of resources, manpower, timescheduling, priorities) are vital to the success of a mission. It is therefore extremely important to thoroughly understand the possibilities of softcomputing methods. An added bonus of the nonlinearly inherent to softcomputing methods is that these systems in addition exhibit an increased robustness compared to classical methods. In studying the various softcomputing techniques such as FS, NN, GA, and OO, it is helpful to imagine a multi-dimensional input-output space, in which we consider a hypersurface with multiple maxima and minima (Fig. 4). FSs and NNs can approximate such a nonlinear input-output relation by combining either a small number of single rules and using very simple basisfunctions (FS), or by just using one type of function (NN).

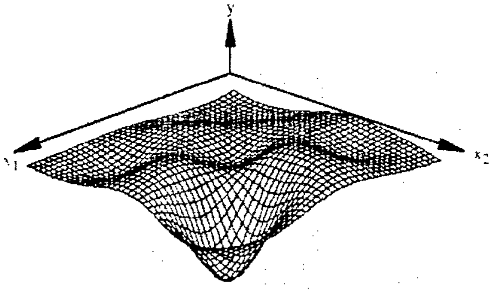


Figure 4 *A hypersurface in 3D space may represent a nonlinear input-output relation (x,y) or a complicated search space with many nearly degenerated optima.*

Basically a FS subdivides the (generally real-valued) variables of the input space in a small set of (overlapping) patches using so-called membership functions (MF), thus bringing down the number of states significantly. Next the nonlinear I/O relation is approximated by defining a rulebase for each of these input states. In a process called 'defuzzification' the fuzzy-valued output is converted to a real output value.

NNs approximate the desired relationship by using sigmoid or radial (Gaussian) basis functions that are weighted, shifted and otherwise modified by varying their synaptic weights in order to achieve the desired approximation. If we imagine the multidimensional space as *search* space, then we can view the output as a kind of performance or *"fitness"* function, measuring the error from some ideal functional behavior on *local* optimization.

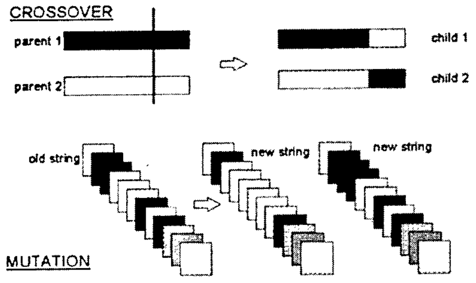


Figure 5 *The basic operations in an genetic algorithm: cross-over of two parent chromosomes forming two children (top) and mutation representing a (rare) stochastic process that randomly flips a gene on a single chromosome.*

GAs are ideally suited to search for a *global* optimum. The key concept is in a genetic optimization routine is the *representation* of the characteristics in a chromosome, a random reproduction process (*cross-over*, Fig. 5) and the *selection* of the 'best' chromosomes from the offspring to produce the next generation. GA has some distinct advantages compared to classical optimization schemes. Its prime advantage is that of fast convergence to near the global optimum. Near the global optimum the converge is however generally very slow. The global searching capabilities are not limited to smooth or simple convex structures: GAs do not require gradients to exist and just rely on a smart representation of the problem onto the chromosomes and a suitable fitness function.

A relatively young branch of softcomputing techniques is that of ordinal optimization. Like in GAs the key issue is here that one has to find a global optimum in a generally complex and large search space. There are two key differences from GAs, that gives OO a place of its own among softcomputing techniques:

1) OO aims to reduce NP-complete searches to polynomial heuristic searches; in other words OO tries to *formalize heuristics*.
2) OO explicitly allows for measurement uncertainty (stochastic observation errors) in the evaluation of the performance of a certain solution.

The basic idea behind various types of heuristic search is that of the 20/80 rule; of which many examples exist, *e.g.*: It takes 20% of time to achieve 80% performance; for the remaining 20% performance one has to spend 80% of his time. Another well-known example is the so-called "birthday paradox". The probability that 2 persons in an arbitrary group of 25 people have their birthday on the same day is > 0.5. Starting from such empirical observations Ho [15] developed the concepts of '*goal softening*' and '*ordinal optimization*'. OO can be intuitively understood by observing that it is generally much easier to determine 'order' instead of 'value' (*e.g.* it is easier to determine that A > B is true than it is to evaluate (A - B)). The concept of goal softening can be visualized by replacing the condition to be satisfied in the optimum by the much easier to fulfill condition of finding an optimum *close enough* to the true optimum.

In the following subsections we will now introduce the different softcomputing methods.

3.1 Fuzzy Logic

Fuzzy logic, fuzzy sets, and fuzzy measures are the basic concepts of FSs. Originally developed as a mathematical theory to model vague, imprecise notions, one of the first applications of FSs was in control (Fig. 6). At a first glance this may appear strange because there is nothing vague or imprecise in control engineering. In contrast one must first fuzzify the measured system inputs to be able to apply the theory of fuzzy sets. The reason for the success of FSs in the domain of control engineering is mainly the capability to absorb human (operator) experience in the form of "rules of thumb", and the capability to encapsulate all the essential knowledge to operate the fuzzy controller in a small set of fuzzy rules. An added advantage of these systems proved to be the robustness of such controllers: the nonlinear controller with its overlapping membership functions could accurately approximate the desired control surface (fuzzy controller = universal optimal approximator). The basic application of FL is through fuzzy (approximate) reasoning: fuzzy control may be viewed as an application of fuzzy approximate reasoning to control (Fig. 7). The key application areas of FSs in information science and engineering are: expert systems, control, feature extraction, and pattern recognition. Recently FS have also been developed in quite different disciplines, *e.g.* medical diagnosis, psychology, economy, management and operations research. In the following we will discuss the essential features of a FS, as illustrated in fuzzy control (FC) and see how FSs model the behavior of a skilled operator instead of modeling the system to be controlled. FC is more aimed at taking actions given certain conditions. The basic idea behind FC is that of *partitioning* the input variable space into a finite number of *overlapping* partitions and defining for each of these partitions a typical output state. The formulation of this definition is in the form of linguistic rules of the type:

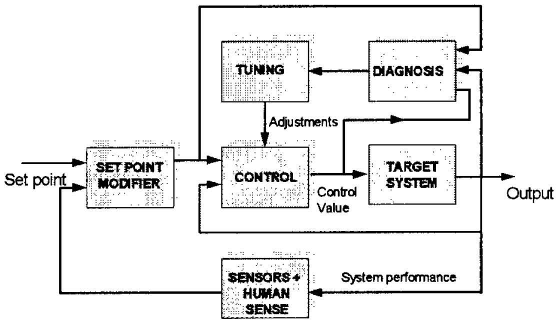


Figure 6 *Fuzzy control can be used in many different ways: Apart from the proper control process, fuzzy logic can be used to diagnose the performance of the controller and fine-tune it, to modify the setpoint and to merge human observations with sensor data.*

"IF (x_1 is Large) AND (x_2 is Small) THEN (y is Negative_Small)"

Here x_1 and x_2 represent fuzzy input variables and y is a fuzzy output variable. Fuzzy variables take linguistic values such as "Negative Large", "Positive Small","Zero", and "Positive Medium". Each of these linguistic values is represented by a membership function μ, *i.e.* a function that is almost everywhere ≡ 0 except for a finite interval, its so-called support, where the function takes positive values ≤ 1 (see Fig. 7). In order to apply this fuzzy rule base it is necessary to fuzzify the crisp (real-valued) input variables. The fuzzification process can be implemented in many ways, but basically it means that the degrees of membership (*i.e.* the values of the MFs $\mu_A(x)$ that are ≠ 0) are associated with the rules having a rulepart of the form "IF (x is A)". In the case that more than one input needs to be considered we must determine the resulting activation of the rule from these degrees of membership. For this purpose a so-called t-norm operator must be selected. Examples of commonly used t-norms are the minimum operator MIN, used by Mamdani:

$$\text{Rulestrength } w_{ij} = \text{MIN}_j\,(\mu_j(x_j)) \quad \text{with } j = 1,2.....k,$$

and the Product-operator Π:

$$\text{Rulestrength } w_{ij} = \prod\nolimits_j (\mu_j(x_j) \quad \text{with } j = 1,2.....k.$$

Finally after aggregating the inputs with the knowledge represented by the rules, the outputs can be determined from the rule strengths and the output membership functions by defuzzifying the

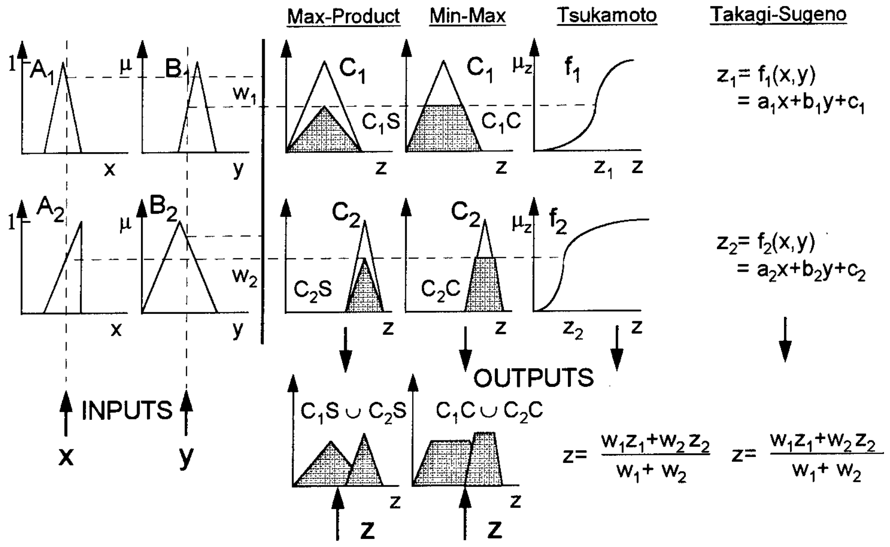


Figure 7 *Four different ways of fuzzy approximate reasoning can approximate different controllers z=f(x,y).*

outputs according to *e.g.* the center-of-gravity method as depicted in Fig.7 for the calculation of the output value z.

3.2 Neural Networks

Artificial neural networks are abstractions of the biological neural networks that constitute the brain. A biological neuron consists of dendrites, a cell body, and an axon (Fig. 8). The connections between the dendrites and the axons are called synapses. Electric pulses are generated in sensory cells (biological sensors) or in neighboring neurons and arrive on the synapses. The cell body operated on these inputs and fires a pulse, *i.e.* outputs an electrical charge on the axon, if the sum of all inputs exceeds a certain threshold. This basic mechanism is copied from the biological system to build an *artificial* neural network (henceforth abbreviated as NN), though excluding the time component: instead of a firing (repetitive) pulse, the output lasts as long as the weighted sum of the input exceeds the threshold. (Fig. 9).

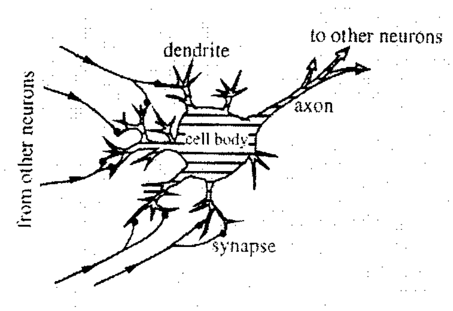


Figure 8 *A biological neuron: inputs from senses and other neurons end in the synapses. The cell body processes these signals and decides to fire, i.e. produce a series of electrical pulses. These are transferred to other neurons via the axon.*

Over time many NN architectures have been developed. In general a neural structure consists of a finite number of inputs connected to the input-layer of neurons and a finite number of output neurons. Between these lie one or more layers of so-called 'hidden' neurons. The idea is that the NN is trained by adapting the weights of the individual neurons so as to replicate the (input, output)-pairs in the training data set. This training can be achieved in two different ways *by supervised learning*, or alternatively, by *self organization*. In supervised learning a training set is available and the learning algorithm adjusts the neuron weights so as to match the desired input-output characteristics. The most frequently used learning algorithm in this category is the backpropagation algorithm. In contrast, unsupervised learning is characterized by a mechanism that changes synaptic weights according to the input values of the network. The output characteristics are therefore determined by the network itself. Examples of self-organization are 1) Hebbian learning in which a weight w_{ij} of a neuron i and an input x_j is increased if the output y_i fires, by an amount $\Delta w_{ij} = \alpha y_i\, x_j$, where α represents the *learning rate* and 2) competitive learning, where all weights are modified of the unit that generates the largest output ('the winner takes all'). An example of such a self organizing competitive NN is Kohonen's self organizing feature map.

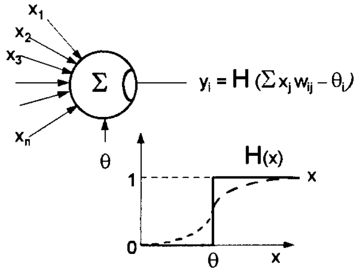


Figure 9 *An artificial neuron forms a linear combination of the inputs x and uses a nonlinear function if the input exceeds the threshold θ.*

One of the most popular learning algorithms is the *backpropagation* algorithm (BP). In the BP algorithm the difference between the desired and actual output of the neural network is backpropagated to modify the weights of all nodes involved in generating the difference. In this sense NN learning is equivalent with finding the global minimum (smallest error) of the error hyperplane in the space spanned by all the weights of the NN.

3.3 Genetic Algorithms

Genetic algorithms are searching for optimization procedures inspired by models of biological evolution. Key features of these so-called evolutionary computation are

1) representation: the coding of the problem under consideration onto chromosomes, *i.e.* strings of numbers (often bits) that code the properties;
2) the definition of an initial population of those chromosomes;
3) the application of biological-inspired random operations such as cross-over (Fig. 5) and, to an extent mutation to generate a new population from the original one;
4) the existence of a "fitness"-function that attributes to each chromosome a fitness value on the basis of which a selection is made of the 'best' chromosomes of the new generation. The 'not so fit' chromosomes are discarded from the population. This is a example of the principle of ordinal optimization to be discussed in the next section. In Fig. 10 a typical computation cycle is shown.

Genetic algorithms are relatively easy to code and converge, dependent on the 'goodness' of the chromosomes representation fairly quick in a number of cases to *near* the global optimum. The introduction of the mutation operator offers a way to escape out of local traps by the random creation of new chromosomes. The implementation of GA's can be very efficient due to the parallel nature of the algorithm: there is no preferential order in which chromosomes should be selected so cross-over can be applied on many chromosomes in parallel.

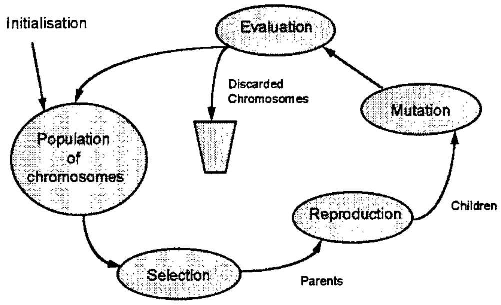


Figure 10 *A generation cycle in a GA: After evaluation the best chromosomes are added to the population and the rest discarded.*

A definite drawback of a GA is the statistical nature of the search with its inherently slow convergence O(1/√n), compared to deterministic methods. Therefore it is of great importance to include as much as possible a priori information in the representation of the chromosomes and into the fitness function. Several other implementations of GAs exist, such as evolutionary programs (EP). EPs are similar to GAs except that mutation is the *only* operator in EP to provide a new generation of chromosomes, thus reflecting the influence of environment (boundary conditions, limitations) rather than that of the parents. In biology this difference is called *phenotype vs. genotype.*

3.4 Ordinal Optimization

Ordinal optimization plays a special role in softcomputing, because the method is not so much an alternative to other softcomputing methods but instead provides insight in the foundations of heuristics that is often used in taking decisions. At various stages of a mission decisions have to be taken in such a way as to optimize the overall goal of the mission. Without exaggeration it can be stated that it is extremely difficult for a human (and even more so for a machine) to really verify that such a decision is in fact optimal. Although the literature of optimization is huge, much of the mathematical analysis is concerned with continuous-differentiable functions, so that calculus-based methods can be used to find an optimum. In most of these methods it is necessary to be able to calculate a gradient or a derivative of a functional in order to find the optimum through a "steepest descent" method, often carried out in a successive approximation approach. This is in fact also true in the case of supervised learning in a NN. Although basically decisions in a neuron can be represented by a step- or Heaviside function of the form: if $\Sigma\ w_{ij}\ x_j \geq \theta_i$ output = +1 else 0 (see Fig. 9), where x_j= jth input synapse of neuron i and w_{ij} = weight associated with the i-th synapse, and θ_i = the threshold, in practice backpropagation requires a one-to-one correspondence between input and output and therefore no discontinuities. It is because of our limited mathematical toolbox (especially in the realm of discontinuous functions, difference equations and discrete optimization) that these calculus-based methods are desired and in act even required to find an optimum solution with analytical means. In order to force solvability, discrete problems are often 'smoothed' and it is hoped that the solution constructed in this way is a good-enough approximate solution of the real problem.

In practice a number of additional real-world difficulties are introduced by boundary conditions, complex geometries and other constraints. These can often be expressed as (in)equalities. Finally we are often confronted with insufficient data, so that some kind of uncertainty modeling has to be done. The introduction of uncertainty into the modeling presents a huge problem in practice, especially if the uncertainty is of a statistical nature (*e.g.* observation accuracy, or sensor noise) and one does not have the possibility (or time) to average over a sufficient number of observations, by which the observation errors can be reduced to an acceptable level. This is often the case in military observation systems. The key issue here is minimizing the estimation error, thus tightening the confidence level of the estimates and convergence to the 'true' optimum. There are a number of problems associated with the calculus-based optimization: 1) discrete-event dynamic systems cannot be treated this way. 2) in real systems it is often very difficult, if not impossible, to prove that the calculated (local) optimum is the desired *global* optimum.

Global minima are therefore only found by running an optimization procedure for multiple starting points (and proving that there are not too many relative optima), or by being able to show that the response surface is convex. In systems of practical interest, *e.g.* in NN this is more the exception than the rule: there we are confronted with an extremely high number of nearly degenerate minima in the energy surface, so that finding the global optimum is virtually impossible. Other examples are the identification of two adjacent frequencies in spectral analysis, combinatorial problems such as finding the shortest way connecting N sites (Traveling Salesman Problem) or scheduling problems such as minimizing production delays, or discrete parameter design. All these problems are difficult because of their enormous search space and the only way to find approximations of the optimum is by running many simulations. In addition they are NP-complete, *i.e.* the time needed to find a solution increases exponentially with the size of the problem. In order to find approximate solutions to such programs one is forced to use heuristics, rules of thumb, and ad hoc methods to achieve some kind of global optimization. Both the difficulty to take into account *all* local detail and the necessity to arrive at a solution in real time have induced a novel softcomputing approach. The justification of the ideas presented by Ho [15,16] is that humans manage reasonably well in making real-world decisions despite the NP-completeness of these problems and the insufficient knowledge.The following example is taken from Ref. [16] and illustrates the basic steps:

Consider, for example, that we have 200 ordered alternatives to evaluate. We blindly pick 12 alternatives out of these 200 and ask "what is the probability that among the 12 picked alternatives there is actually at least one alternative that is in the top-12?" The surprising answer is 0.5 ! If the number 12 is changed to 35, then the probability of finding a "good" alternative is close to one in the above statement. The implication of this is that even in the absence of any knowledge, one can dramatically reduce the number of alternatives one has to evaluate to narrow the search for "good" choices.

The central idea behind the previous statement is that of *ordinal optimization*: the idea that the relative order (instead of the cardinal value) of the performance of various alternatives in a general decision problem is quite robust with respect to estimation noise. The number of true top-r alternatives in the set of estimated top-r alternatives can be quite substantial even in the face of very large estimation errors in the performance value of the alternatives. In the above example or randomly picking alternatives, the equivalent estimation noise has infinite variance. If, on the other hand, the variance is not infinite, *i.e.*, there is some bias in favor of the actual good alternatives (however slight), then we can only improve the odds and help to narrow down the search. This is the core of the probabilistic justification of using heuristics in complex decision problems.

3.5 How to combine softcomputing with classical methods

All softcomputing methods have their proper application areas and it is impossible to even approximately describe how they can generally be combined. It is also clear that in contrast to the original introduction of these systems as competing and independent developments, one observes nowadays a convergence towards intelligent, hybrid systems [17], where NNs are combined with FS to achieve a certain level of adaptability. Also in more complex systems it is customary to distinguish a hierarchy of levels of autonomy and depending on the fuzziness of the goals and the uncertainty and ambiguity present in the observations, the resulting system is implemented as a mixture of classical and softcomputing approaches. It is therefore important to have a general idea of how methods could be combined in a meaningful way.

TABLE 1: A hierarchy of modeling techniques. It should be noted that all these methods can in principle be combined with each other, but that a higher one in the hierarchy (*i.e.* more analytical) is preferred over a lower one.

Model	Theory	Properties	Uncertainty	Speed
Analytical	Calculus	high precision continuous,global, numeric	no	off-line, fast
Rule-based	Fuzzy Logic	discrete, finite precision, local, structural, symbolic	yes, incomplete and ambiguous	on-line fast
Lookup table	Neural Network	learning, numeric, local, black box, unstructured	yes, noise can improve training	learning: off-line slow
Global optimization	Genetic algorithms	numeric, global, evaluation function	simulated annealing	slow
Ordinal Optimization	Ordinal Optimization	probabilistic, global and local	goal softening	fast

For the novice in this field it can be very bewildering to see how different authors solve the same type of problem with very different methods, each with its own merits, but it is almost impossible to compare the performance of these methods without a deep understanding of the underlying processes and actually re-doing the experiment. Although still many publications are devoted to one of the softcomputing techniques discussed here, without even giving a rationale why the selected method is preferred in this case, it should be pointed out that the so-called 'intelligent hybrid systems' gain rapidly in importance and aim at an integral approach of all techniques available. This is the line of thought that we also adhere. Without trying to review all combinations of NN, FS, GA, and OO that have been published we will try to give a guideline according to which the various softcomputing techniques might be applied: (see Table I).

Following the ordering of Table I, we start with conventional, often linearized modeling of the problem at hand. The advantage of such methods is that there is a well-developed body of mathematical methods available for this type of problems and even in the nonlinear case analytical expressions, conservation laws, and other relations can readily be derived. These methods have the great advantage that they are fast to evaluate (because analysis is essentially off-line), provide very accurate data and also provide insight in the underlying mechanisms: They allow us to parametrize environmental variables and allow us to explore their effect on the solution. In the absence of continuity and high precision data, or if the underlying problems are too complicated to model, solve analytically, or calculate numerically, it pays to approximate the 'exact' truth by trading in precision for speed of calculation. It has become only recently clear that precision can be very costly and that it may be much more efficient to use underlying structural knowledge of the type "IF X increases a bit THEN Y decreases strongly". It is in this context that fuzzy rule bases become important. At one hand they allow us to deal with 'difficult' details of classical analytical systems, at the other they provide us with a means to 'fuse' human operator experience with physical observations and mathematical models based on differential equations. The accuracy (*i.e.* input-resolution) diminishes in such systems, but the overall approximation of the observed system behavior increases. If structure is completely absent (at least the underlying structure cannot be recognized), but a sufficiently large 'training data set' *i.e.* (input, output) pairs is available, it is worthwhile to model the system at hand as a black box. This method has some drawbacks: the concept of a blackbox is not appealing to the scientist because one is never sure that a training set is of the correct size. Nevertheless NNs can provide a powerful method in extracting patterns in *e.g.* image recognition. Once the NNs are trained sufficiently (which is a slow process due to the statistical $O(1/\sqrt{n})$ performance), a NN provides a fast on-line lookup-table for connecting inputs and outputs. In addition it is theoretically possible to approximate *any* mapping $\Re^n \rightarrow \Re^m$ with arbitrary precision, provided enough training data are available. In the absence of numeric training data, it may be possible to find an optimal approximation of a good representation and a suitable evaluation function. It should be noted that good representation is always an essential step in finding a solution to a problem. The advantage of a GA approach is that discrete optimization is possible. Unlike in the case of *e.g.* NNs there is no need for gradients to exist and optimization is only on the basis of a suitable evaluation function. Complex boundary conditions may be taken into account and convergence to near the global optimum is generally fast. Finally if the search space for the optimal solution becomes prohibitively large (as is case of large NP complex problems), ordinal optimization (OO) may provide a solution path. Ordinal optimization tries to formalize common heuristics by softening the goal and finding the optimum within acceptable accuracy and probability limits through the use of even the slightest global information on the topology and structure of the search space, without suffering from the slow performance imposed by the $O(1/\sqrt{n})$ limit.

Each method has its particular strengths and weaknesses. Depending on the availability of training data and a priori structured knowledge a model may be developed that approximates the observed behavior sufficiently accurate. The penalty for not using the available a priori information is that the system has to discover these structures itself at the price of slow convergence (law of large numbers). It is well-worth considering to decrease the required accuracy because this generally results in a substantial reduction of computation time. Finally it should be noted that, like in all modeling problems, representation often is the key to solving a problem: without the proper algebra it is often virtually impossible to find a solution, whereas with a suitable representation the solution may even look trivial.

4. CONCLUSION

In this review we have discussed a number of basic softcomputing techniques and have indicated how they may be combined to form intelligent hybrid systems suitable for optimizing missions. Key aspects of these systems such as their capability to handle uncertainty, their adaptability, and their robustness make such systems of great interest for future mission systems. The latter are expected to evolve from human decision support systems towards unmanned, highly autonomous robotics systems that should be capable of reaching their goal, even in adverse circumstances and with scarce resources. The increasing complexity of these tasks and the limited capacity to include more computing power into the mission platform (ignoring the relevance of including more computing power because of the NP-completeness), it is of great importance that mission systems benefit from modern developments in softcomputing, thereby increasing their robustness to cope with new and unexpected situations. Although progress in many other technologies will certainly be needed to achieve the ambitious goals set for future (un)manned missions, we are convinced that softcomputing technology is one of the key technologies to achieve this aim.

5. ACKNOWLEDGMENT

The author wishes to thank Ms. Ellen J.M. Tieland for her help in preparing the manuscript.

6. REFERENCES

[1] J. Llinas and E. Waltz, Multisensor Data Fusion, ArtechHouse, Norwood, MA.,1990.

[2] D. Hall and R. Linn, “A taxonomy of multisensor data fusion techniques”, Proc. 1990 Joint data fusion symposium, vol 1, 593-610, 1990.

[3] C.B. Weaver, Ed., “Sensor Fusion”, Proc. of SPIE, vol 931, Orlando, FL, 1988.

[4] P.S.Schenker, Ed., “Sensor Fusion, Spatial reasoning and scene interpretation”, Proc. of SPIE, vol 1003, 1988.

[5] C.B. Weaver, “Sensor Fusion II”, Proc. of SPIE, vol 1100, Orlando, FL, 1989.

[6] P.S.Schenker, “Sensor Fusion II, human and machine strategies”, Proc. of SPIE, vol 1198, Philadelphia, PA, 1989.

[7] A.J. van der Wal, “Application of fuzzy logic control in industry”, Fuzzy Sets Syst 74, 33-41, 1995.

[8] A.J. van der Wal, ”Fuzzy Logic: Foundations and Industrial Applications”, ed. D. Ruan, Kluwer, Chapter 14, 275-311, 1996.

[9] D.Ruan, Z. Liu, L. Van den Durpel, P.D’hondt, and A.J. van der Wal, “Progress of Fuzzy Logic control applications for the Belgian Nuclear reactor BR1”, Proceedings EUFIT ‘96, Aachen vol2, 1237-1241, 1996.

[10] A.J. van der Wal and D. Ruan, Proc. JCIS 97, Research Triangle Park 136, 1997.

[11] G. J. Klir and B. Yuan, Fuzzy sets and fuzzy logic, Prentice Hall, New Jersey, 1995.

[12] A. P. Dempster, “Upper and lower probabilities induced by a multivalued mapping”, Ann. Math. Statistics, 38, 325-339,1967.

[13] G. Shafer, A mathematical theory of evidence, Princeton University Press, Princeton, 1976.

[14] K. Leszczynski, P. Penczek, and W. Grochulskki, “Sugeno's fuzzy measures and fuzzy clustering”, Fuzzy Sets Syst., vol 15 147-158, 1985.

[15] Y.-C. Ho and M.E. Larson, "Ordinal optimization approach to rare event probability problems", J. DEDS 5, 281-301, 1995.

[16] Y.-C. Ho, "Heuristics, rules of thumb, and the 80/20 proposition", IEEE Trans. on Automatic Control 39 (5), 1025-1027, 1994.

[17] For an up to date review: "Intelligent hybrid systems: fuzzy logic, neural networks, and genetic algorithms", Ed. D. Ruan, Kluwer Academic, Boston, 1997.

LEARNING FUZZY RULES FROM DATA

Robert J. Hammell II
US Army Research Laboratory
Aberdeen Proving Ground, MD 21005, USA

Thomas Sudkamp
Department of Computer Science
Wright State University
Dayton, OH 45435, USA

SUMMARY

Fuzzy models have been designed to represent approximate or imprecise relationships in complex systems and have been successfully employed in control systems, expert systems, and decision analysis. Classically, fuzzy models were built from human expertise and knowledge of the system being modeled. As systems have grown more complex it has become increasingly difficult to construct models directly from domain knowledge of the system. Recently, learning algorithms have been investigated to construct fuzzy models by developing fuzzy rules through analysis of training data. This research presents a hierarchical architecture for fuzzy modeling and inference that learns the underlying fuzzy rules from training data. The effects of increased granularity and dimensionality on the performance of the system are illustrated and discussed, along with techniques to minimize the adverse impacts of this increased complexity. A short discussion of ongoing research aimed at allowing the combination of human expertise with fuzzy learning strategies is also included.

1. INTRODUCTION

Fuzzy set theory provides a formal method for modeling complex systems. In classical modeling, system relationships are expressed as mathematical functions. As the systems of interest become more complex, it is increasingly difficult to develop mathematical models directly from knowledge of the system. This is due not only to the complexity of interactions within the system, but perhaps based on an incomplete knowledge of the system operations as well. A fuzzy model uses a set of fuzzy rules to provide a functional approximation of the relationships of the underlying system. The popularity of fuzzy models is attributable to their ability to represent complex, imprecise, or approximate relationships that are difficult to describe in precise mathematical models.

Historically, fuzzy rules have been obtained by knowledge acquisition from experts. Recently, learning algorithms have been employed to analyze a set of training examples and build the rule base(s) of a fuzzy model. Techniques for building fuzzy models from training data were presented by Wang and Mendel [1], Kosko [2], and examined in Sudkamp and Hammell [3]. The accuracy of the resulting rule base is affected by several factors, such as the input domain decompositions, the number of examples in the training set, and the precision of the training data.

Learning rules from training examples admits the possibility that the resulting rule base may be incomplete. That is, there may be a possible input for which no action is specified. This problem is overcome by the use of completion algorithms. Completing a rule base uses the existing rules and interpolation to produce rules for the undefined configurations [4].

This paper presents an examination of the process of, and problems associated with, learning fuzzy rules from data. A hierarchical architecture for constructing fuzzy models is introduced and the effectiveness of the systems constructed from this architecture are examined. The problems within fuzzy learning algorithms caused by increased granularity and dimensionality are illustrated and discussed along with two techniques that may be useful for minimizing the impacts of these problems. Ongoing research involving the combination of domain expertise with fuzzy learning algorithms is briefly presented.

2. FUZZY RULE-BASED SYSTEMS

A fuzzy set represents approximate or vague information over a domain. The membership function μ for a fuzzy set assigns a grade of membership ranging from 0 to 1 for all elements in the domain. For domain U, a fuzzy set A over U is defined by a membership function $\mu_A{:}U\rightarrow[0,1]$ where $\mu_A(x)$ denotes the membership grade of x in A. The membership grade $\mu_A(x)$ indicates the degree to which x satisfies the predicate A.

The ***height*** of a fuzzy set A is the supremum of the

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

membership grades of the elements in A. A normal fuzzy set has a height of 1; otherwise, the set is subnormal. The *support* of a fuzzy set A is the crisp set containing all elements of the domain that have non-zero membership in A. The *core* of a fuzzy set consists of all elements with a membership grade equal to 1. In triangular fuzzy sets the core is a unique element and is referred to as the midpoint.

A two-input, one-output fuzzy rule-based inference system with inputs from domains U and V and output in W is shown in Figure 1. Fuzzification may be used to transform the input based on noise in the data source or on the degree of precision desired for the analysis. Defuzzification performs the opposite transformation by converting the output fuzzy set C' over W to a singleton $c \in W$.

The first step in the design of such a fuzzy inference system consists of defining the terms to be used in the antecedents and consequents of the rules. This specifies the language of the rule base, and is accomplished by decomposing the input and output domains into families of fuzzy sets. A decomposition of a domain U is a set of fuzzy sets $A_1, ..., A_n$ over U wherein the union of the supports of the fuzzy sets completely covers U. Figure 2 shows the domain [-1,1] decomposed into 5 triangular fuzzy sets.

The antecedents and consequents of the rules are determined by the decompositions of the input and output domains, respectively. The decomposition of a domain U is assumed to form a fuzzy partition of U. A fuzzy partition satisfies

$$\sum_{i=1}^{n} \mu_{A_i}(u) = 1$$

for every u in U.

The combination of triangular membership functions and the fuzzy partition requirement ensures that any input will have non-zero membership in at most two fuzzy sets (A_i and A_{i+1}), and that the input will have membership of $\geq .5$ in at least one fuzzy set. In this paper, all domains are assumed to be normalized to take values from the interval [-1,1] and the domain decompositions consist of a partition by triangular fuzzy sets.

The domain knowledge of a fuzzy model is encapsulated by its set of fuzzy rules. A fuzzy rule with two antecedents has the form 'if X is A and Y is B then Z is C' where A and B are fuzzy sets over the input domains U and V, respectively and C is a fuzzy set over the output domain W. Thus, a fuzzy rule is used to linguistically represent relationships between domains. For example, 'if speed is *fast* and distance is *medium* then apply brake *medium*' is a fuzzy rule that might be used in a brake pressure control system, where *fast* is a fuzzy set describing speed, *medium* a fuzzy set over distance, and *medium* in the consequent a fuzzy set over pressure. The linguistic variables are used to approximately characterize the values of variables as well as their relationships. The combination of linguistic variables and fuzzy set theory provides a formal system for representing and manipulating imprecise or approximate quantities.

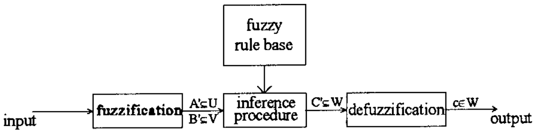


Figure 1. Fuzzy Inference System

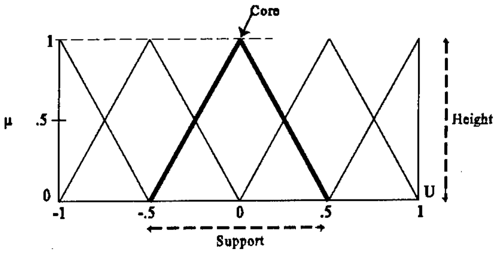


Figure 2. Decomposition of [-1,1]

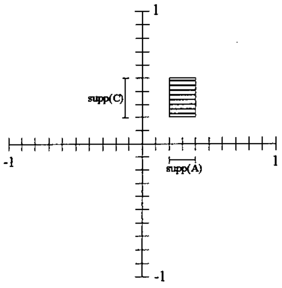


Figure 3. Fuzzy Rule Patch

The region in which a rule provides information for inference is defined by the supports of the fuzzy sets. Figure 3 illustrates the region of influence for the rule 'if X is A then Z is C'. The supports of A and C describe a fuzzy relation patch which bounds the area covered by the rule. This is called the *locality principle* of fuzzy approximation. As more fuzzy sets are used, the region covered by each rule shrinks. Thus, the domain decompositions define the granularity of the approximations.

A fuzzy associative memory (FAM) is an l-dimensional table where each dimension corresponds to one of the input universes. The i'th dimension of the table is indexed by the fuzzy sets that comprise the decomposition of the i'th input domain. Consider a fuzzy model with input domains U and V, and output domain W, as depicted in Figure 1. Let $A_1, \ldots, A_m$ be a partition of U and $B_1, \ldots, B_n$ be a partition of V. A fuzzy rule base for such a system consists of rules of the form 'if X is A_i and Y is B_j then Z is $C_{k_{ij}}$' where $C_{k_{ij}}$ is a fuzzy set over the output domain W. The FAM representing this rule base has the form

	A_1	A_2	...	A_{m-1}	A_m
B_1	$C_{k_{11}}$	$C_{k_{21}}$	...	$C_{k_{(m-1)1}}$	$C_{k_{ml}}$
⋮	⋮	⋮		⋮	⋮
B_n	$C_{k_{1n}}$	$C_{k_{2n}}$	...	$C_{k_{(m-1)n}}$	$C_{k_{mn}}$

where the consequent of a rule with antecedent 'if X is A_i and Y is B_j' is entered in the (i,j)'th position.

A fuzzy rule base consists of a series of rules that trace a fuzzy function from the input domain to the output domain. A one-input one-output system with input domain decomposition $A_1, \ldots, A_7$ and output domain decomposition $C_1, \ldots, C_7$ is used to show the interpretation of fuzzy inference as approximation. A rule base is defined by the 1-dimensional FAM

A_1	A_2	A_3	A_4	A_5	A_6	A_7
C_1	C_2	C_3	C_4	C_3	C_2	C_1

Figure 4 shows the supports of the fuzzy sets in the domain decompositions along the horizontal and vertical axes. The boxed areas indicate the regions in the space that are influenced by each rule. The parabolic shape of the overlapping regions suggests that this may be a fuzzy approximation of the function $f(x) = -(x^2)$.

Fuzzy inference compares the input with the antecedent of a rule to produce the response indicated by that particular rule. The output is obtained by summarizing the responses indicated by each individual rule. As mentioned above, a fuzzy rule 'if X is A then Z is C' represents a functional relationship between input A and output C. The area covered by such a rule is bounded by the supports of A and C. For example, the shaded lower right-hand rectangle in Figure 4 is the region associated with the rule 'if X is A_7 then Z is C_1'.

The role of a fuzzy model is to determine an appropriate output response based on the input to the system. The inference technique determines the degree to which each rule affects the outcome. For the experimental results given in this paper, Mamdani style inference and weighted averaging defuzzification are used to produce the output; this algorithm will be described using a one-input one-output inference system where the domain decompositions are triangular with evenly spaced midpoints. The input universe is subdivided into triangular regions $A_1, \ldots, A_n$ and the output domain is decomposed by the fuzzy partition $C_1, \ldots, C_m$ with associated midpoints $c_1, \ldots, c_m$. The restrictions on the decomposition make inference in such a system very straightforward and efficient. Assume that A_i and A_{i+1} are the two fuzzy sets providing nonzero membership for an input value x with associated FAM entries C_r and C_s, respectively (there will only be two for a one-input system). Using weighted averaging defuzzification, the result z specified by input x is given by

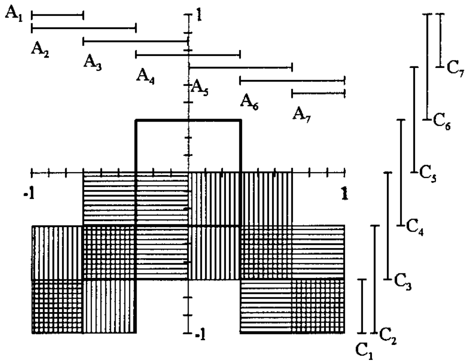


Figure 4. Fuzzy Inference as Approximation

$$z = \frac{\mu_{A_i}(x)c_r + \mu_{A_{i+1}}(x)c_s}{\mu_{A_i}(x) + \mu_{A_{i+1}}(x)} \tag{1}$$

Rewriting (1) in terms of the midpoints produces

$$z = \frac{x(c_s - c_r) + a_{i+1}c_r - a_i c_s}{a_{i+1} - a_i} \tag{2}$$

where c_r and c_s are as defined above and a_i and a_{i+1} denote the midpoints of A_i and A_{i+1}, respectively. Equation (2) shows that the result is completely determined by the rule base and the midpoints of the triangular decomposition of the domains. As mentioned earlier, each of the input and output domains are assumed to be the interval $I = [-1,1]$. When the input is precise, a fuzzy model and weighted-averaging defuzzification defines a real-valued function $\hat{f}: [-1,1]^k \rightarrow [-1,1]^t$ where k and t are the dimensions of the input and output domains respectively.

3. LEARNING FUZZY RULES

Wang and Mendel [1] introduced an algorithm for generating FAM entries from training data. For a one-input system, the training data consists of a set of input-output pairs $T = \{(x_i, z_i) \mid I = 1, \ldots, k\}$ where x_i is an element from the input domain and z_i is the associated response. The

Wang and Mendel algorithm, herein called FLM, is described below. More detailed examinations of FLM and comparisons with other learning algorithms can be found in [3] and [5].

The FLM algorithm is defined as follows: A training example (x_i, z_i) that has the maximal membership in A_j is selected from T. If more than one example assumes the maximal membership, one is selected arbitrarily. The fuzzy rule 'if X is A_j then Z is C_r' is constructed where the consequent C_r is the fuzzy set in the output domain decomposition in which z_i has maximal membership. If z_i has maximal value in two adjacent regions ($\mu_{C_r}(z_i) = \mu_{C_{r+1}}(z_i) = .5$), then the consequent C_r is selected.

The generation of a rule with antecedent 'if X is A_j' requires at least one training example with membership .5 or greater in the set A_j. Learning a large number of rules may require a large set of training data, especially with multi-dimensional input. An inference system with five input domains, with each decomposed into five regions, produces a FAM with 3125 rules. If the training set is obtained by sampling an operational system, a suitable training example may not be encountered for all FAM entries. Entries left undefined by FLM are filled using rule base completion ([3],[4]) which is examined in more detail in the subsequent example and in Section 3.2 below.

To enhance the performance of models constructed using supervised learning from training data a two-level architecture was designed by modifying the FLM algorithm [6]. The FLM algorithm is extended to utilize an analysis of the error between the approximating function $\hat{f}$ and the training data. The first step of the FLE (Fuzzy Learning with Error analysis) algorithm is to construct the initial approximation $\hat{f}$ with FLM. The second step then re-uses the training data to refine the approximation. This is done by producing a second FAM that defines a function $\hat{f}_e$ which is used to approximate the error between the training data and $\hat{f}$.

The training set T_e used to learn $\hat{f}_e$ is obtained from the training data T and the initial approximation $\hat{f}$. An element in T_e represents the difference between an original training example and the approximation produced by $\hat{f}$; the training set is $T_e = \{(x_i, z_i - \hat{f}(x_i)) \mid (x_i, z_i) \in T\}$.

To define the fuzzy associative memory to approximate the error function, called the EFAM to distinguish it from the original FAM, it is necessary to identify the input and output domains of the error function. The input domain is [-1,1], the same as the approximating function $\hat{f}$. The output of the error function, however, takes values from a smaller interval than the domain of the original FAM. To allow for the largest possible error, the EFAM size is set to one-half of the length of the support of the largest fuzzy set in the FAM output domain decomposition. For an output domain decomposition of 5 equally spaced triangular fuzzy sets, the support of each such fuzzy set is an interval of length .5; the associated EFAM output domain is the interval of [-.25, .25].

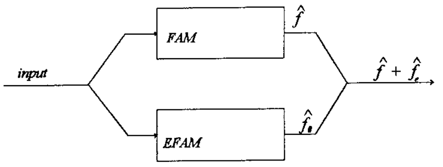


Figure 5. Two-level Architecture

Once the domain decompositions have been selected, learning the function $\hat{f}_e$ follows the FLM algorithm using T_e as the training set. The last step in the creation of the EFAM is to use completion to fill in any missing entries.

The approximation produced by the FLE algorithm uses the two FAMs as illustrated in Figure 5. An input x is processed by each of the FAMs. The resulting values $\hat{f}(x)$ and $\hat{f}_e(x)$ are added to produce the single overall system output. The following example details the steps in the process of constructing a two-level system.

Example: This example illustrates constructing a two-level system to approximate the function $f(x) = x^2$ from training data $T = \{ (-1,1), (-.4,.16), (0,0), (.5,.25), (.7,.49), (1,1) \}$. Note that the training set consists of precise data; each pair has the form $(x, f(x))$ with no noise introduced into the function value.

The first step is to construct the approximation $\hat{f}$ from T. In this example, the input and output universes for the FAM are decomposed into the five equally spaced triangular fuzzy regions with midpoints -1, -.5, 0, .5, and 1 as shown in Figure 2. The FLM algorithm produces the FAM

A_1	A_2	A_3	A_4	A_5
C_5	C_3	C_3	C_3	C_5

which provides the approximation $\hat{f}$.

Observe that the x values in both training pairs (.5,.25) and (.7,.49) belong to region A_4 the most. However, since .5 has a higher membership value in this region than .7, the corresponding z value of .25 is used to determine the consequent for the rule 'if X is A_4...'. Also note that training set T completely covers the input regions so no rule base completion is necessary.

If the process were to stop here the result would be that of the original Wang and Mendel algorithm. To illustrate how the addition of the EFAM increases the accuracy of the approximation, the table below shows the approximations $\hat{f}(x)$ provided by the FAM for the original input values from the training set. The second line of the table is what the actual output should be $(x=f(x))$; the third line of the table is the approximation provided by the FAM; the last line of the table provides the error in the approximation.

$x =$	-1	-.4	0	.5	.7	1
$z = f(x)$	1	.16	0	.25	.49	1
$\hat{f}(x)$	1	0	0	0	.4	1
$z - \hat{f}(x)$	0	.16	0	.25	.09	0

The remaining steps continue the FLE algorithm by augmenting the FAM with an EFAM. The second step is to build the training set T_e for the construction of the EFAM. From above, the $(x, \hat{f}(x))$ pairs are (-1,1), (-.4,0), (0,0), (.5,0), (.7,.4) and (1,1). Combining a training example $(x, z) \in T$ with $\hat{f}$ produces the example $(x, z - \hat{f}(x))$. The error training set $T_e = \{$ (-1,0), (-.4,.16), (0,0), (.5,.25), (.7,.09), (1,0) $\}$.

To begin the construction of the EFAM, it is necessary to define the output domain and the decompositions. Let the input domain [-1,1] be decomposed into seven regions as shown in Figure 6. The output domain, however, will not be the interval [-1,1]; it is determined by using one-half of the length of the support of the largest fuzzy set in the FAM output domain decomposition. For the input domain decomposition of 5 equally spaced triangular fuzzy sets as used above, the EFAM output domain is the interval [-.25,.25]. Let this interval also be decomposed into seven regions as illustrated in Figure 7.

The EFAM constructed using T_e is

A_1	A_2	A_3	A_4	A_5	A_6	A_7
E_4		E_6	E_4		E_5	E_4

Note that two entries in the EFAM are unfilled; each represents an input for which no action is specified. Overcoming this problem motivates the use of completion algorithms. Completing the EFAM uses the existing rules and interpolation to produce rules for the undefined configurations.

The completion algorithm used in this research is a region growing technique often used in image segmentation [8]. Empty cells in the EFAM that border nonempty cells are filled by extending the values in the neighboring cells. Since the 'growing' only occurs on the boundaries of filled regions, the procedure must be repeated until the entire table is filled. Region growing is the most local of a family of completion algorithms examined in [3].

Using region growing, the completed EFAM is

A_1	A_2	A_3	A_4	A_5	A_6	A_7
E_4	E_5	E_6	E_4	E_4	E_5	E_4

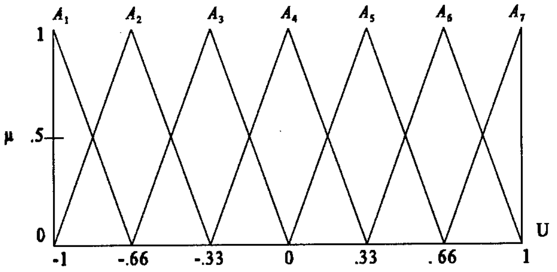


Figure 6. EFAM Input Decomposition

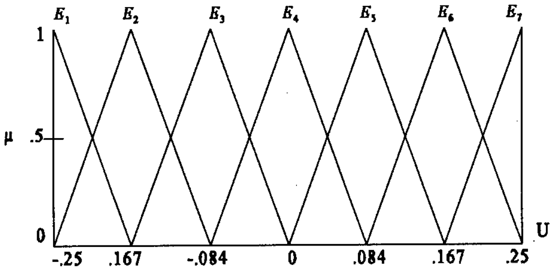


Figure 7. EFAM Output Decomposition

The improvement obtained by incorporating the error function $\hat{f}_e$ into the approximation is shown in the table below. The values in the table are the error, at the training points, in the approximation of the function $f(x) = x^2$ using $\hat{f}$ alone compared with that of $\hat{f} + \hat{f}_e$.

$x =$	-1	-.4	0	.5	.7	1
$z - \hat{f}(x)$	0	.16	0	.25	.09	0
$z - (\hat{f}(x) + \hat{f}_e(x))$	0	.01	0	.25	.02	0

The region growing technique allowed the rule base to be completed so that a reasonable output could be produced for all possible inputs. The completion process for region A_5 was limited to arbitrarily choosing either E_4 or E_5; if E_5 had been picked then the error for x = .5 would have decreased to .17 with all other results remaining the same. Also, it should be observed that for input x = -.4 the approximation used the 'grown' consequent of region A_2, producing a significant decrease in error over the original FAM-only system.

The hierarchical system presented here has been further modified to provide an adaptive system. An initial two-level system is created from the training set as outlined above and, once operational, system performance feedback is used to modify the FAM and EFAM rule bases to allow adaptation to a changing environment. The hierarchical adaptive systems have been tested on three types of adaptive scenarios: continued learning, gradual change, and drastic change. Details of the adaptive algorithm along with experimental results can be found in [7].

In addition to illustrating the algorithmic steps, the above example shows other aspects of the FLE algorithm as well. One point is the increase in accuracy provided by using the error analysis to construct the EFAM. Another very important detail is the use of completion, especially within the EFAM. These two features of the algorithm will now be discussed in more detail.

3.1 Increased Accuracy

The example shown above illustrates how adding the EFAM to incorporate the error function $\hat{f}_e$ into the approximation can improve the overall accuracy of the system. Experimental analysis was conducted to compare FAM-only and FAM-EFAM systems over a set of one and two-input target functions utilizing various system configurations and training set sizes. The average and maximal error of the approximations obtained from a test set were recorded and compared. The results of these experiments were reported in detail in [6].

The experimental analysis results showed that the incorporation of $\hat{f}_e$ reduced the average error in all cases. For the maximal error, there was a single case where the original algorithm (FLM) outperformed the two-level architecture (FLE). The data show that the two-level architecture is clearly superior to the single-FAM systems. The construction of the FAM-EFAM system requires only one additional pass through the training data, maintaining the efficiency of the FAM learning approach.

The data also indicate that the selection of the number of regions in the EFAM should be influenced by the number of training examples. With a small training set, the construction of the EFAM results in many cells unoccupied. The accuracy of the completion algorithm decreases as the amount of cells it must fill grows. Consequently, it is preferable to use fewer regions in the EFAM when only a small number of training instances is available.

While the experiments showed the improvement that can be obtained by using an EFAM with the FAM learning algorithm, the improvement did not come for free. The reduction in error was achieved by adding a second rule base. This begs the question of whether or not it would be better to simply use more regions in the FAM of a single-FAM system. On a strictly theoretical level free of resource limitations, the answer is yes. The single FAM system with an unlimited number of regions in the domain decompositions produces a universal approximator of real-valued functions [3]. That is, for any real-valued function f there is a FAM that will approximate f to within any predetermined degree of accuracy. Using a learning algorithm to build such a FAM, however, may require an inordinate or unobtainable amount of training data.

The work reported in [6] illustrated that with a limited amount of training data the two-level architecture can outperform the single FAM systems in several important ways. First, the FAM-EFAM system requires fewer training examples than a single FAM system to reach a desired level of accuracy. Both types of systems are universal approximators but the FLE algorithm generally produces comparable performance with a smaller training set.

As noted above the increase in performance for the FLE algorithm is obtained at the expense of adding the rules that comprise the EFAM. The experiments also showed, however, that the two-level architecture generally outperforms single FAM systems even when the same total number of rules is used. That is, the two-level systems produce more accurate approximations even when the total rules (FAM rules + EFAM rules) is equal to or less than the total rules in a single FAM system.

Lastly, with a fixed number of training examples, a two-level system can outperform a FAM-only system regardless of the number of regions used in the FAM of the single-level system. Since single FAM systems are universal approximators of real-valued functions, a FAM architecture can be constructed that has the ability to approximate the target function to within any desired degree of accuracy. This is done by decomposing the input and output domains into sufficiently small regions. However, increasing the number of regions requires a corresponding increase in the size of the training set needed for the FLM algorithm to learn the function. Thus, there will come a point when a fixed size training set will no longer support larger FAMs. That is, completion becomes less accurate as more empty FAM entries have to be approximated from the few that are filled. The experiments showed that with a fixed amount of training data a simple FLE system may be able to outperform a single-FAM system of any size.

3.2 Completion

When constructing a fuzzy rule base with the FAM learning algorithms presented in this paper it is possible that an incomplete rule base may result. That is, there may be insufficient information in the training set to produce rules for every possible input condition. If this should happen, the FAM would have one or more cells with no assigned value. Such a situation is not acceptable for many systems since this indicates a combination of inputs for which the resulting action is undefined.

The value of completion was clearly demonstrated in several experiments reported in [4]. For example, with a target function of $x/2$, 50 randomly selected training examples were not sufficient to build a 25-rule FAM. However, completing the resulting incomplete FAM produced an approximation whose performance was identical to that of a 25 rule FAM built with 200 training examples (no completion needed with 200 examples). Thus, in this example, completion allowed the construction of a system that performed as well as one built with four times the training data.

In the two-level architecture, the experimental data showed that a small FAM may be used to get a crude approximation of the desired function and then an EFAM with more regions can be used to "tune" the final output. Using a large number of regions in the EFAM to provide this finer granularity increases the possibility that a limited size training set will not contain enough information to fill all cells. As such, completion was not generally used in constructing the FAM but was essential in constructing the EFAM.

The philosophy behind completion is that of analogy and similarity. Classically, analogy hypothesizes an answer based on the nearest similar case. It was not the goal of this research to develop new and improved completion strategies. As such, the technique used follows the strategy of region growing commonly employed in image segmentation [8]. As applied here, this produces an interpolation over the rule set rather than over the data (training) set. Using interpolation in the context of fuzzy set theory has been called plausible inference [9].

Much more detail about the theory, implementation, and examples of completion as used in this work can be found in [3] and [4]. Specifically, [3] compares two completion techniques, outlines the general notion of approximation with similarity relations, and introduces a general completion algorithm. This information is carried further by presenting two applications of rule base completion in [4].

4. GRANULARITY AND DIMENSIONALITY

The majority of successful applications of fuzzy systems have a limited number of input domains. In particular, systems built for control applications generally have two or three input variables. As increasingly sophisticated systems are modeled, the complexity of the FAM will grow accordingly.

The complexity of a FAM can be affected in two different ways. One way is by increasing the number of fuzzy sets in the decomposition of an input domain; this determines the *granularity* of the system which thereby defines the space of approximating functions that may be realized [3]. Granularity increases as the fuzzy sets are made smaller, thus growing in number. The second way to increase the complexity of a FAM is to add additional input variables. The number of input domains for the system determines the *dimensionality* of the system.

As seen in the previous discussions regarding EFAM construction, increasing the granularity usually causes a corresponding need for an increase in the amount of training data. Increasing the dimensionality of a system will also cause a need for a larger training set. In addition to possibly requiring more training data, increases in dimensionality or granularity also affect the off-line learning of the rules and the run-time performance of the resulting system. These two aspects are examined below along with the presentation of a method for reducing the impact of dimensionality.

4.1 Run-Time Considerations

The execution cycle of a fuzzy system consists of the acquisition of input, determination of the appropriate set of rules, the evaluation of the rules, and the aggregation of the results. The final value, either fuzzy or precise depending upon the application, is then returned as the appropriate response or action.

During operation, the input is received and the next step is to determine the set of rules that are applicable for the input. For well-defined domain decompositions, especially the triangular decompositions with the fuzzy partition constraint, hash functions may be used to make this determination independent of the number of terms in the decomposition [12]. Thus, the granularity of the system has little effect on this aspect of the system. For multi-dimensional FAMs, the work required for determining the applicable rules increases linearly with the number of dimensions; the overall set of indices is the Cartesian product of the indices of the individual dimensions.

The introduction of additional input dimensions, however, may have an exponential affect on the time needed to evaluate the rules. For a precise input value and a triangular domain decomposition, there are typically two fuzzy sets in which the input has a nonzero support; the sole exception to this is when the input has membership value of 1 in a single fuzzy set. For an n-dimensional system, there are 2^n applicable rules. If the input is fuzzy, the set of applicable indices in each domain increases, exacerbating the growth in the number of rule evaluations required.

Finally, the amount of work required in the aggregation of the results of the individual rules is determined by the number of rules that were evaluated. Thus, the impact of increasing the number of rules in each dimension (increased granularity) can be limited by the use of hash functions, but increased dimensionality will adversely affect the performance of the system.

Although increased granularity does not affect the run-time performance of the system, it has important consequences on the modeling capabilities. Since the number of fuzzy sets determines the family of realizable approximating functions, increasing the granularity presents the possibility of overfitting the training data. Overfitting occurs when a learning algorithm attempts to match minute variations within the training data rather than producing a generalization from the data. Intuitively, one may consider the ability to generalize from the training data to vary inversely with the granularity of the input domains.

4.2 Rule Generation Considerations

The primary effect on the rule generation process caused by increased dimensionality or granularity is the corresponding need for a larger training set. A larger training set will, of course, impact the amount of computer memory required to execute the learning algorithm as well as increase the time required to construct the rule set. These drawbacks are minimal, however, especially given the processing speed and available memory of modern computers and the fact that the rules will be learned off-line. The main drawback is simply the need for more training data and the negative impacts on system accuracy if the required amount of data is not available.

The generation of a rule by the learning algorithm requires at least one training example in the region defined by the supports of the fuzzy sets in the antecedent of the rule. If the training set is small or the rule base is large, there may be FAM regions that do not contain a suitable training example. As discussed in the example above and in Section 3.2, completion can be used to interpolate over the rule base and fill in missing rules. The ability of the completion process to accurately build the missing rules is dependent upon the density of the rules that are learned, which is dependent upon the density of the training data. Completion can be used to minimize the effect of a moderate increase in granularity. Results from extended completion experiments can be found in [3,4].

Even with completion, increases in granularity can impact the performance of the resulting system. In theory, an increase in granularity permits a more precise model to be constructed. This is only true, however, if there is sufficient training data to exploit the precision afforded by the model. Table 1 illustrates the impact of increased granularity on the performance of the hierarchical learning algorithm. The model configuration *n/m* indicates the number of fuzzy sets in the decomposition of the FAM and EFAM, respectively. The target function is a three-input function; thus, a 5/9 system has 125 FAM rules (5 x 5 x 5) and 729 EFAM rules (9 x 9 x 9).

The first set of data in Table 1 indicates that when the number of EFAM regions is increased the precision of the model decreases slightly. Other experiments have demonstrated that the point at which the precision of the model declines is dependent upon the amount of available training data. The effectiveness of completion depends upon the ratio of regions that are filled compared to those that must be filled by the completion algorithm. Increasing the number of EFAM regions while maintaining a fixed size training set alters this ratio.

The second set of data in Table 1 illustrates that increasing the number of training examples for a fixed configuration produces a more accurate model. This reflects the dependence of the learning algorithm upon the data rather than on the interpolation technique.

Model	# Examples	Max Err	Avg Err
Fixed Number of Examples			
5/9	5000	.250	.016
5/13	5000	.290	.017
5/17	5000	.350	.019
5/21	5000	.201	.019
5/25	5000	.209	.221
Fixed Configuration			
5/9	5000	.250	.016
5/9	7500	.189	.015
5/9	10000	.189	.013
5/9	20000	.127	.009

Table 1. Increasing EFAM Granularity

Increased dimensionality also leads to a larger FAM rule base which in turns drives the need for more training data as well. A 5/25 system for a two-input function has 25 FAM and 625 EFAM rules; the same system for a three-input function has 125 FAM rules and 15,625 EFAM rules.

The easiest way to demonstrate the effect of dimensionality on the hierarchical learning algorithm is to add a redundant dimension to the system. That is, choose a two-dimensional target function $f(x,y)$ and, from f, construct the three-dimensional target $f'(x,y,z) = f(x,y)$. Under these conditions the difference in the effectiveness of the learning algorithm with targets f and f' can be attributed directly to the additional input dimension.

Table 2 provides the results of approximating $f(x,y) = (x+y)/2$ and $f'(x,y,z) = (x+y)/2$. For the 5/25 system configuration, the three-dimensional system generated with 50,000 training examples was less accurate than the two-dimensional approximation produced with 5000 examples. This same pattern is exhibited by the 9/25 and 13/25 systems. Similar results were found in the experiments on models of one and two input dimensions [3]; an order of magnitude increase in the number of training examples was required to produce models of approximately equal precision.

4.3 Active Regions and Dimension Reduction

The effects of multiple input dimensions on FAM systems were clearly illustrated by the results in Table 2. Consequently, strategies need to be developed that reduce the effects of increasing dimensionality on the efficiency and data requirements of the learning algorithms. In this section two such techniques are presented.

In complex systems, there may be combinations of

Model	# Examples	Max Err	Avg Err
Approximation: $f(x,y,z) = (x+y)/2$			
5/25	5000	.209	.221
	10000	.209	.012
	20000	.106	.009
	50000	.073	.007
9/25	20000	.081	.009
	50000	.052	.007
13/25	50000	.076	.007
Approximation: $f(x,y) = (x+y)/2$			
5/25	500	.132	.015
	1000	.088	.009
	5000	.043	.005
9/25	5000	.027	.005
13/25	5000	.049	.006

Table 2. Two and Three Dimensional Results

potential inputs that do not represent feasible system configurations. When this occurs, the corresponding regions of the input space need not be covered by a model. Moreover, there may be distinct and identifiable regions of the input space in which the system is constrained to operate. These disjoint areas in the input space will be referred to as the *active regions*.

The above conditions, which occur for some but not all problems, were encountered in the development of a fuzzy model to predict the propagation loss of a signal [10]. The model used five characteristics of the topography, the signal, and the relationship between the transmitter and the receiver to produce an estimate of the signal loss. A rule base was generated from a set of 100 training examples.

The training data for this model were concentrated in several distinct regions of the input space. The particular region could be determined by the value of a single input variable. Due to the properties of this application, a large model covering the entire five-dimensional input space was not required; a set of four-dimensional FAMs was sufficient to cover the active regions of the input space.

Because of the small number of training data, the identification of the active regions was straightforward for the propagation loss model. In more complex problem domains and with larger training sets, the dependencies may not be easily discernable. For this reason, the following approach for the location of active regions and dimension reduction is proposed.

The first step is to use the training set to identify areas of high activity in the input space. If the training set is obtained from sampling an operating system, then the selections should provide a representative distribution of the input processed by the system. Clustering is then used to group the training data and define regions of the input space. If the analysis produces compact and separable clusters, these are identified as the active regions of the system. The criteria for the degree of compactness and separability required to identify a distinct active region depends upon the application and the dimensions of the input space.

When the active subregions of the input space have been identified, a FAM will be generated to cover each such region from the training data in that region. The entire model will then consist of a supervisor that receives the input and a set of localized FAMs. The action of the supervisor is to direct the input to the appropriate FAM for evaluation. By only generating rules to cover the active regions in the input space, the requirement for training data should be reduced.

The second technique is to identify and eliminate unnecessary dimensions. As in the case of the propagation model, the localized FAMs may not require all of the dimensions of the input space to produce an appropriate response. In fact, the target function and experimental data shown in Table 2 is an example of this type of behavior.

The test for the contribution of the *i*th input dimension begins by constructing the n-dimensional FAM for the region. This FAM defines an n-dimensional function $\hat{f}$. The n-1 dimensional FAM is then built by deleting the *i*th dimension from the training set, creating an approximating function $\hat{f}_i$. The values of the two approximations are compared on the training set. The maximal difference between the approximations

$$er_i = \max\{|\hat{f}(x_1,\dots,x_n) - \hat{f}_i(x_1,\dots,x_{i-1},x_{i+1},\dots,x_n)|\}$$

is recorded. Following a 'leaving-one-out' strategy, this process is repeated for each input dimension. The dimension with minimal difference, call it I, is considered for removal. If er_i is less than a predetermined threshold, the reduced FAM obtained by omitting the it dimension will be used. The threshold for removal of a dimension is based on the degree of precision in the training data and the precision required from the system. This process can be iteratively repeated with the potential of removing additional dimensions from the domain.

The strategy underlying the dimension deletion procedure is similar to the selection of attributes for testing in the construction of decision trees and the removal of features in the derivation of classification rules [11]. The hierarchical system consisting of a supervisor and localized FAMs of smaller dimension than the input space combine to reduce the effect in the growth of the number of rules that must be considered in high dimensional systems.

5. COMBINING EXPERTISE AND LEARNING

Fuzzy models are currently either constructed using information provided by domain experts or generated using learning algorithms with training data, but not by both. The fuzzy sets in the rules developed by learning algorithms are often selected based on the efficiency of calculations or based on the distribution of the training data. Being data driven, the algorithms construct precise rules with each rule having a limited range of applicability over the input space.

Experts, however, decompose the problem into subdomains determined by the similarity of the input conditions. The resulting rule base generally consists of a small number of rules with each rule covering a large number of situations. Thus, the granularity and precision of rules provided by domain experts differ considerably from those of rules generated from training data.

Ongoing work is investigating the relationships between granularity and specificity in fuzzy rule bases. Specificity measures, also know as possibility measures, indicate the degree to which a fuzzy set designates a unique item in the universe. A strategy is being developed for learning fuzzy rules that optimizes granularity based on a desired degree of specificity. In addition, the approach should be able to be used to combine rule bases of various degrees of granularity.

A main ingredient of the approach is to use Takagi-Sugeno-Kang (TSK) rules [13] instead of Mamdani style rules. A TSK rule has the form 'if X is A_i then $g_i(x)$' where $g_i(x)$ is a crisp function from U to V. The output for an input $x \in U$ is the average of the $g_i(x)$'s weighted by the degree of membership in A_i. Frequently the consequent functions are linear functions of the input variables and the combination of rules of this form produces a piecewise linear function from U to V.

A preliminary algorithm has been developed to construct a fuzzy approximation given training data and a specification of acceptable specificity for each point in the domain. The intuition behind the algorithm is to begin a regression line fitting the data and extend it as long as all the training instances fall within the region defined by the line and the specificity requirement. Experiments to test the efficacy of the approach are the next step.

6. CONCLUSIONS

The results and techniques presented in this paper represent preliminary investigations into the robustness of the double-FAM learning algorithm. The objective of the experiments was to determine if the advantages of the hierarchical model carry over into more complex problem domains. The creation of robust learning algorithms will ultimately require a combination of techniques. These techniques will be dependent upon the available training data, the basic properties of the system being modeled, and the degree of accuracy required from the model. The combination of domain expertise and fuzzy learning may be an essential part of future models of complex systems.

REFERENCES

1. Wang, L. and J.M. Mendel, "Generating Fuzzy Rules from Numerical Data, with Applications", Technical Report USC-SIPI-169, Signal and Image Processing Institute, University of Southern California, Los Angeles, CA 90089, 1991.

2. Kosko, B., "Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence", Prentice Hall, Englewood Cliffs, NJ, 1992.

3. Sudkamp, T. and R.J. Hammell II, "Interpolation, Completion, and Learning Fuzzy Rules", IEEE Transactions on Systems, Man, and Cybernetics, 1994, 24, 2, pp.332-342.

4. Sudkamp, T. and R.J. Hammell II, "Rule Base Completion in Fuzzy Models", in Fuzzy Modeling: Paradigms and Practice, Norwell, MA, Kluwer Academic Publishers, 1995.

5. Hammell II, R.J., "A Fuzzy Associative Memory for Fuzzy Inference and Adaptivity", Ph.D. thesis, Wright State University, Dayton, OH, 1995.

6. Hammell II, R.J. and T. Sudkamp, "A Two-Level Architecture for Fuzzy Learning", Journal of Intelligent and Fuzzy Systems, 3, 4, 1995, pp. 273-286.

7. Hammell II, R.J. and T. Sudkamp, "An Adaptive Hierarchical Fuzzy Model", Expert Systems with Applications, 11, 2, 1996, pp.125-136.

8. Haralick, R.M. and L.G. Shapiro, "Image Segmentation Techniques", Computer Vision, Graphics, and Image Processing, 29, 1985, pp. 100-132.

9. Prade, H., "Approximate and Plausible Reasoning", in Fuzzy Information, Knowledge Representation and Decision Analysis, Oxford, UK, Pergamon Press, 1984.

10. Hayes, M., "The Development of the Fuzzy Propagation Loss Model, Version 2", Technical Report, CSC Corporation, 1986.

11. Chiu, S., "An Efficient Method for Extracting Fuzzy Classification Rules from High Dimensional Data and Its Aerospace Applications", *in* "Proceeding of the Int'l Workshop on Breakthrough Opportunities for Fuzzy Logic", Tokyo Institute of Technology, 1996.

12. Sudkamp, T., and R.J. Hammell II, "Scalability in Fuzzy Rule-Based Learning", Information Systems Journal, accepted Oct 97, to be published.

13. Takagi, T., and M. Sugeno, "Fuzzy Identification of Systems and Its Application to Modeling and Control", IEEE Transactions of Systems, Man and Cybernetics, 15, 1985, pp. 329-346.

Real-Time Object Structuring and Real-Time Simulation for Future Defense System Engineering

K. H. (Kane) Kim and Chittur Subbaraman
Dept. of Electrical & Computer Engineering
University of California
Irvine, California 92697, U.S.A.
Kane@Ece.Uci.Edu

Abstract

The authors believe that the *real-time object structuring* technology and the *real-time simulation* technology are among the most important computer technologies needed to support future engineering of advanced defense systems. Although the real-time object structuring technology has not yet been established in a mature form, there is no alternative to it with respect to enabling economic and reliable design of complex computer based application systems. It facilitates not only reuse of modules tested in earlier applications but also production of system designs that are easy to understand and modify. The other important technology, the real-time simulation technology, supports an advanced mode of simulation in which the simulator modules are designed to show the same timing behavior that the simulation targets do. High-fidelity real-time simulators of application environments can thus be used in effective validation of real-time computer-based control systems. Such validation has great cost and flexibility advantages and is becoming further attractive due to the on-going decrease in the costs of developing real-time simulators. The authors present one version of the real-time object structuring technology and the related real-time simulation technology along with some laboratory experiences in prototyping small-scale defense applications by using these technologies.

1. Introduction

Large-scale command-control systems needed in future air defense are the most challenging subject for the computer-based system technologists who are about to enter a new millenium. In order to meet the public expectations of the development efficiency, the costs, and the reliability of such systems, the system engineering methods different from those used in the past must be employed. The authors believe that the *real-time object structuring* technology and the *real-time simulation* technology are among the most important real-time computer based system engineering technologies to be established.

First, although the real-time object structuring technology has not yet been established in a mature form [Wor94, Wor96, Wor97], there is no alternative to it with respect to enabling economic and reliable design of complex distributed and parallel computing application systems. It facilitates not only reuse of modules tested in earlier applications but also production of system designs that are easy to understand and modify.

Secondly, real-time simulation is an advanced mode of simulation in which the simulator modules are designed to show the same timing behavior that the simulation targets do [Kim96b]. High-fidelity real-time simulators of application environments can thus be used in effective validation of real-time computer-based control systems. Such validation has great cost and flexibility advantages and is becoming further attractive due to the on-going decrease in the costs of developing real-time simulators. The cost reduction is being realized not only by the reduction in the hardware costs of distributed and parallel computer

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

systems but also by the development of easy-to-use object programming tools and stable real-time operating systems. This authors believe that the real-time simulation technology will advance rapidly in coming years.

This paper presents briefly one version of the real-time object structuring technology and the related object-structured real-time simulation technology as examples of those which will be increasingly studied by the defense computing technology development community in coming years. This paper also presents some laboratory experiences in prototyping small-scale defense applications by using these technologies.

The two technologies needed in future engineering of challenging defense systems are also needed in future automation of industry and social infrastructures. However, advanced defense systems present the greatest challenges to the technology development community. Therefore, investment by the defense research and development community in these technology developments is expected to have very positive impacts on the commercial sector dealing with industry automation and social infrastructure development applications.

2. The needs for real-time object structuring in engineering of challenging defense systems

In this section, several major reasons why new real-time object structuring technologies are needed in engineering of challenging defense systems are discussed.

2.1 Design complexity

The authors believe that large-scale command-control systems needed in future air defense will present greater design complexity than any other artifacts designed so far by human beings have presented. Among the factors that contribute to the design complexity are the following imposed on the systems:

(1) the stringent response time requirements,
(2) the requirements for cooperative decision-making by widely distributed dynamic subsystems, and
(3) the requirements for fusion and noise-filtering of fuzzy sensor information that is large in volume, generation rate, and variety.

Such complex systems cannot be built to possess an acceptable level of quality without using futuristic structuring techniques that have the potential of being highly effective in producing easily understandable designs of real-time distributed and parallel computing systems.

2.2 Exploitation of advanced technologies developed in the commercial sector

Defense systems needed are much smaller in quantity in comparison to the typical products in commercial markets. Experiences have shown that when the defense community develops computer technologies in complete separation from the computer technology developments in the commercial business sector for a substantial length of time, the results are generally overshadowed by similar but more reliable technologies emerging from the commercial sector.

Over the past 15 years *object-oriented* (OO) design approaches have become a common practice in the development of non-real-time business data processing software due to the modularity, generality, and natural abstraction benefits that the OO approaches bring in [Dah72, El190, Boo91, Rum91, Sel94]. On the other hand, OO-structuring has had minimal impacts in real-time computer systems (RTCS) engineering in contrast to its pervasive use in non-RTCS engineering. This means that much of the capabilities existing in the vast business data processing software field is currently not utilized in development of RTCS's such as those needed in defense applications. It also means that the currently practiced RTCS engineering process and the current real-time application software themselves take peculiar forms unfamiliar to the vast main-stream software engineering community. The consequence is the poor economy of scale in RTCS development and the relatively low reliability of the software products except in cases of small-scale simplistic phase-locked loop control types of applications.

2.3 Reuse of modules tested in earlier applications

Module reuse is of great importance as a means of keeping future software engineering costs down. OO programming tools that emerged in the last decade represent a major advance in facilitating module reuse in non-RTCS engineering fields. To facilitate module reuse in RTCS engineering, an extension of the conventional object structure and supporting tools that are effectively applicable to RTCS engineering and reengineering must be established first.

2.4 Reliability

Development of complex RTCS's in esoteric forms, which has been widely practiced up to now, is very unlikely to lead to reliable products. When large-scale defense systems have low reliability, consequences can be devastating to large communities. Also, it is impossible to test and validate many defense systems to the same degree of thoroughness that is achieved in testing of commercial non-RTCS's. Therefore, general-form design and easily understandable/analyzable design are of critical importance in producing defense systems. The reliability concerns are then essentially as the concerns on how to conquer the design complexity discussed above in Section 2.1.

2.5 A natural solution: Real-time object structuring

The arguments made above in Sections 2.1 - 2.4 lead to one conclusion. One of the most important technologies that need to be established is an extension of the existing object structuring technology that is effectively applicable to design of RTCS's and supports the following which is called the General-form timiliness-Guaranteed (GG) design paradigm [Kim95a, Kim97a, Kim97b]:

(1) *General-form design*: Future real-time computing must be realized in the form of a generalization of the non-real-time computing, rather than in a form looking like an esoteric specialization. In other words, under a properly established real-time system design methodology, every practically useful non-real-time computer system must be realizable by simply filling the time constraint specification part with unconstrained default values.

(2) *Design-time guarantee of timely service capabilities of subsystems*: To meet the demands of the general public on the assured reliability of future RTCS's in safety-critical applications such as defense applications, there does not appear to be any adequate way but to require the system engineer to produce design-time guarantees for timely service capabilities of various subsystems (which will take the form of objects in object-oriented system designs). Experiences of practicing engineers indicate that testing alone is not sufficient for assuring the level of reliability of RTCS's which the customers have started demanding. It is also known that in general, verifying the full logical behavior of sizable real-time software is not practical. However, the authors believe that verification of the timing behavior is economically feasible and must be pursued. It is actually the timing behavior which presents the biggest difficulties to the system engineer relying on the testing for assuring proper behavior to a reasonable degree. The ease or difficulty of verifying the timing behavior depends on the way time constraints are specified in real-time objects.

Therefore, the essence of the GG design paradigm is to realize real-time computing in a general manner not alienating the main-stream computing industry and yet allowing system engineers to confidently produce certifiable RTCS's for safety-critical applications. Conventional object structures do not have concrete mechanisms for flexible and accurate representation of temporal behavior of complex dynamically changing systems. They need to be extended to support general-form design of RTCS's. Research activities in this direction for establishing extended object structuring technologies, which can be called *real-time object structuring* technologies, have started growing rapidly in recent years [Att91, Ish92, Kim94b, Sel94, Tak92] although most of the views held by other researchers may not have been as idealistic as this authors' view of accepting both the general-form design paradigm and the design-time guarantee of timely service capabilities as the feasible and most appropriate goals.

3. The needs for real-time simulation in engineering of challenging defense systems

Real-time simulation is an advanced accurate mode of simulation in which *the simulation objects are designed to show the same timing behavior that the simulation targets do* [Kim96b]. Efficient real-time simulation technologies are under increasing demands. Two major application areas of such technologies are discussed in this section and both application areas represent many defense applications.

3.1 Virtual reality applications

A growing application field of the real-time simulation is the virtual reality field. A virtual reality environment corresponding to a dynamic physical environment, e.g., a compartment in a moving train or a flying airplane, is not of high quality if the virtual environment does not change at the same tempo at which the physical counterpart changes. To effect precise real-time simulation, the *simulation execution engine* (a computer to run the simulation program) must be of the type that exhibits predictable and dependable timing behavior.

3.2 Cost-effective testing of RTCS's with real-time simulators of application environments

After a control computer system has been implemented, its testing typically involves interfacing it with the application environment and performing test-runs. In the case of a command-control computer system, finding or setting up a proper application environment is very expensive if not impossible. Therefore, it is often inevitable or at least highly useful to connect such control computer systems to a simulator of the application environment for its validation rather than directly proceeding to interface the computer system with the application environment. A highly desirable simulator here is one capable of accurately imitating the timing behavior of the environment, i.e., real-time simulation of the environment.

The environment simulator based testing approach has also other advantages such as great cost and flexibility advantages. Its high flexibility characteristics enables high-coverage testing involving simulation of a large variety of environment conditions. On the other hand, these cost and flexibility advantages are meaningless if the accuracy and precision of the simulation are low. They are meaningful only if accurate true real-time simulation is achieved.

Real-time simulation of complex application environments such as defense application environments requires dependable real-time parallel and distributed computing technologies. The slow maturing of the latter has had the preventive effect on the emergence and wide use of the former. As real-time parallel and distributed computing technologies started growing faster in recent years, the real-time simulation technology has also started showing faster advances. In other words, the costs of developing real-time simulators have started decreasing fast. The cost reduction is being realized not only by the reduction in the hardware costs of distributed and parallel computer systems but also by the development of easy-to-use object programming tools and stable real-time operating systems. This authors believe that the real-time simulation technology will advance even more rapidly in coming years.

3.3 Object structured real-time simulation

In order to support accurate real-time simulation, new approaches to model the simulation targets, especially those which enable multi-fidelity representation of the timing behavior of the simulation target, are needed. In our view, a preferred modeling approach for use in real-time simulation should be of object-oriented (OO) type because the modularity, generality, and natural abstraction benefits of OO approaches in non-real-time conventional simulation have been amply demonstrated in the practicing field [Dah72].

Since existing object models do not possess adequate capabilities for representing the timing behavior accurately, we are again forced to search for a proper extension of the basic object model. As mentioned in the preceding section, the object structure which possesses strong capabilities for representing the timing behavior accurately and varying degrees of precision, is needed to support cost-effective design of real-time embedded computer systems as well. Naturally, finding such extended object models has emerged as one of the most important research issues in the real-time computing field in this decade [Kim95a, Kim97b]. Ideally, a model that is capable of uniformly and accurately representing both real-time embedded computer systems and application environments, is the most desirable.

4. An overview of the TMO structuring scheme

As an example of newly emerging extensions of the conventional object structure that are aimed for supporting RTCS engineering, the real-time object structuring scheme called the time-triggered message-triggered object (TMO) structuring scheme, formerly called the TMO structuring scheme, is reviewed in this section. The TMO is particularly suited for flexible and yet accurate specification of the timing behavior of modeled subjects. An abstract precursor to the TMO was jointly formulated in late 1980's [Kop90] and it has evolved into the TMO structuring scheme with a concrete syntax structure and execution semantics in recent years [Kim94a, Kim94b, Kim96a, Kim97d].

Only a brief overview of the TMO structuring scheme is given in this section. More details can be found in the references. The TMO was devised with the ultimate goal of facilitating the idealistic design paradigm discussed in Section 2, i.e., the GG design paradigm. It facilitates not only production of easily understandable and analyzable designs of RTCS's but also efficient production of real-time simulators of application environments. It thus enables uniform structuring of control computer systems and application environment simulators.

The basic structure of a TMO is depicted in Figure 1. It is an extension of the conventional object and three most important extensions are the following:

(a) *Two clearly separated groups of methods*:

For some methods of a TMO, a real-time clock serves as the mechanism for triggering the method executions. These object methods whose executions are triggered as the clock reaches some values specified at design time, are called time-triggered (TT-) methods or spontaneous methods (SpM's). They are *clearly* separated from the conventional service methods (SvM's) triggered by request messages from clients. The two types of methods in a TMO are different not only in the way their executions are triggered but also in that "actions to be taken at real times *which can be determined at the design time* can appear only in SpM's". Therefore, actions of the type "at constant-clock-value do S" or the type "sleep-until constant-clock-value" can appear only in SpM's.

Triggering times for SpM's must be fully specified as constants in the section called the autonomous activation condition (AAC) section during the design time. The AAC for SpM1 in Figure 1 may be: "for t = from 10am to 10:50am every 30min start-during (t, t+5min) finish-by t+10min", which has the same effect as

{"start-during (10am, 10:05am) finish-by 10:10am",
"start-during (10:30am, 10:35am) finish-by 10:40am"}.

(b) *Basic concurrency constraint* (BCC):

In order to dramatically reduce the designer's efforts in guaranteeing timely service capabilities of TMO's, the execution rule which prevents conflicts between SpM's and SvM's is incorporated. Basically, *activation of an SvM triggered by a message from an external client is allowed only when potentially conflicting SpM executions are not in place.* To be exact, when a message-triggered SvM is not free of conflict with an SpM in accessing the same portion of the *object data store* (ODS), execution of the former method (SvM) must not be allowed in a time zone earmarked for a TT-execution of the latter method (SpM). This restriction is called the basic concurrency constraint (BCC). Therefore, SpM's are given

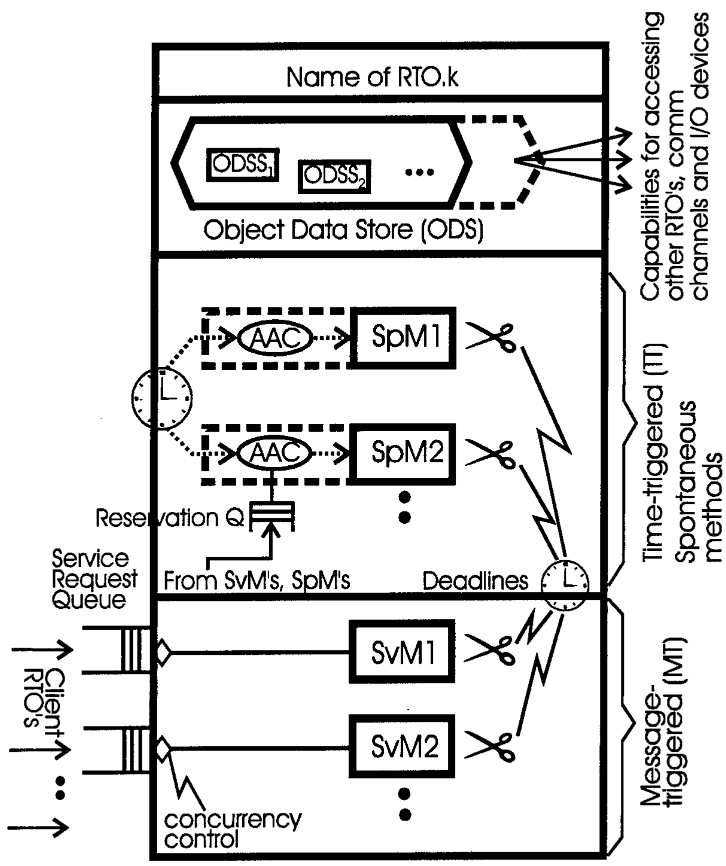


Figure 1. The TMO structure (adapted from [Kim94a])

higher priorities for execution over the SvM's. Note that this BCC does not impose any restriction on concurrent execution of SpM's or concurrent execution of SvM's. Therefore, executions of SpM's are not disturbed by SvM executions and possible triggering times of SpM's are fixed at the design time. At least this makes it very easy to analyze the execution time behavior of SpM's. For example, if a statement of the type "at 10am do S" appears in an SpM, its reliable execution can be easily assured.

(c) For each execution and completion of a method of a TMO, a deadline is imposed;

The first two features (a) and (b) mentioned above make the TMO structure clearly distinguished from other proposed real-time object structures [Att91, Ish92, Kim94a, Tak92].

The designer of each TMO provides a guarantee of timely service capabilities of the object by indicating the *deadline for every output* produced by each SvM (and each SpM which may be executed on requests from SvM's) in the specification of the SvM (and some relevant SpM's) advertised to the designers of potential client objects. Before determining the deadline specification, the server object designer must convince himself/herself that with the object execution engine (hardware plus operating system) available, the server object can be implemented to always execute the SvM such that the output action is performed within the deadline. Again, the BCC contributes to major reduction of these burdens imposed on the designer.

The TMO structuring scheme is effective not only in the multiple-level abstraction of real-time (computer) control systems under design but also in the accurate representation and simulation of the

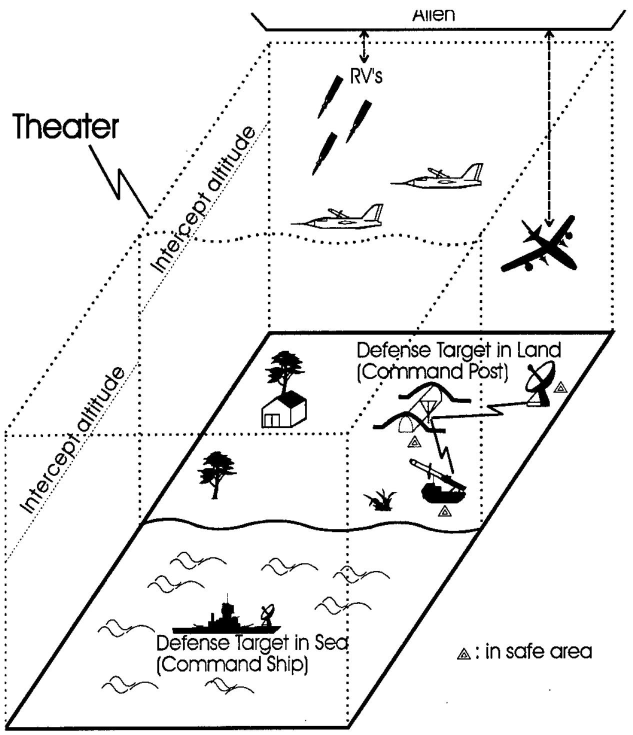


Figure 2 CAMIN application environment: Theater
(Adapted from [Kim96c]

application environments. This uniform structuring presents considerable potential benefits to the system engineers. An illustration of this aspect is given in the next section.

The relationship between the TMO structuring scheme and the conventional process-oriented structuring of real-time concurrent programs is analogous to that between the high level language programming and the assembly language programming.

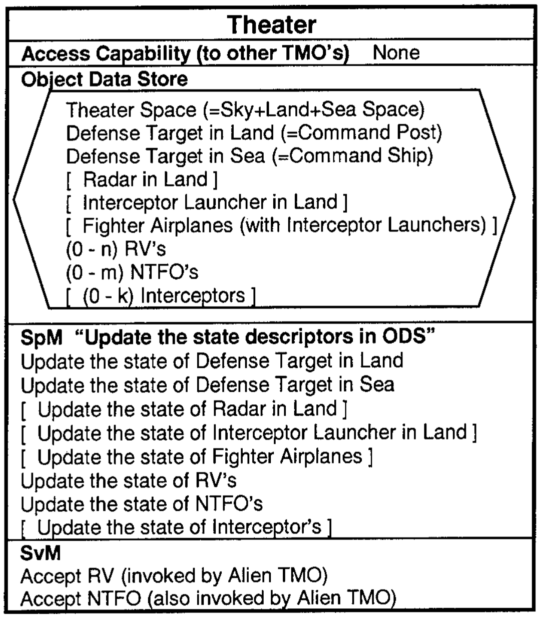
5. TMO based design and simulation: An example

Consider the anti-missile defense scenario depicted in Figure 2. The application environment in this context is a sky+land+sea segment of interest, called the "theater", in which moving objects including a *valuable target to be defended* (i.e., command ship in sea) and *flying objects* of both hostile and non-threatening types, appear and move around. The defense system to be constructed is called the *Coordinated Anti-Missile Interceptor Network* (CAMIN) [Kim96c].

Initially, the high-level requirements are given by the customer who places an order for the CAMIN:
(1) Each reentry vehicle (RV) should be intercepted if it is considered dangerous; and
(2) If it is useful in avoiding the dangers posed by RV's, move the defense target (e.g., command ship) around.

5.1 Step 0: High-level Specification of the Application Environment of the CAMIN as a TMO

Initially, sensors such as radars and actuators such as interceptor launchers (both air-borne and ground-based) do not exist because the system engineer (team) has not decided which types to use. As the first step, the system engineer may describe the application environment of the CAMIN as a TMO depicted in Figure 3 without the components enclosed by square brackets. This TMO is called the *Theater TMO*. The object data store (ODS) of the Theater TMO basically consists of the "state descriptors" for a defense target in sea (command ship), a defense target in land (command post), dynamically varying numbers of RV's and *non-threatening flying objects* (NTFO's) (e.g., commercial airplanes and birds), and the space (=sky+land+sea space) in the theater. The information kept in the Theater TMO is thus a composition of the information kept in all the state descriptors within its ODS.



Theater

Access Capability (to other TMO's) None

Object Data Store

Theater Space (=Sky+Land+Sea Space)
Defense Target in Land (=Command Post)
Defense Target in Sea (=Command Ship)
[Radar in Land]
[Interceptor Launcher in Land]
[Fighter Airplanes (with Interceptor Launchers)]
(0 - n) RV's
(0 - m) NTFO's
[(0 - k) Interceptors]

SpM "Update the state descriptors in ODS"
Update the state of Defense Target in Land
Update the state of Defense Target in Sea
[Update the state of Radar in Land]
[Update the state of Interceptor Launcher in Land]
[Update the state of Fighter Airplanes]
Update the state of RV's
Update the state of NTFO's
[Update the state of Interceptor's]

SvM
Accept RV (invoked by Alien TMO)
Accept NTFO (also invoked by Alien TMO)

Figure 3. High-level specification of the Theater TMO
(Components enclosed by square brackets are chosen by the engineer)

For each *environment object* represented by a state descriptor in the Theater TMO, there is a spontaneous method (SpM) for periodically updating the state descriptor. Conceptually the SpM's in the Theater TMO are *activated continuously* and each of their executions is *completed instantly*. The SpM's can then represent continuous state changes that occur naturally in the environment objects. The natural parallelism that exists among the environment objects can also be precisely represented by use of multiple SpM's which may be activated simultaneously.

The service methods (SvM's) in the Theater TMO are provided as an interface for the clients outside the theater. The only conceivable clients here are the enemy which sends RV's into the theater and the "natural forces" which cause NTFO's (such as commercial airplanes and birds) to enter the theater. Entry of an RV into the theater is represented by an enemy's call for the SvM "Accept RV". Both the enemy and the external forces are represented by a TMO called the *Alien TMO*.

So far, the Theater TMO in Figure 3 has been interpreted as a mere description of the application environment. However, if the activation frequency of each SpM is chosen such that it can be supported by an object execution engine, then the resulting Theater TMO becomes a *simulation model*. The behavior of the application environment is represented by this simulation model somewhat less accurately than by the earlier description model based on continuous activation of SpM's. In general, the accuracy of a TMO structured simulation is a function of the chosen activation frequencies of SpM's.

5.2 Step 1: High-level design of the application environment simulator based on theTMO model

Upon receiving the customer's order, the system engineer will first decide on the set of sensors and actuators to be deployed in the theater. Figure 2 already includes some sensors, i.e., radars which are located both in land and in the command ship, and some actuators, i.e., interceptor launchers which are located in land, in the command ship, and in the fighter airplanes. After the set of sensors and actuators is determined, the Theater TMO in Figure 3 is expanded to incorporate all the components enclosed by square brackets. The ODS now contains the selected sensors (e.g., Radar in Land) and actuators (i.e., Interceptor Launcher in Land with Interceptors). The radar and interceptor launcher loaded on the command ship and another interceptor launcher on the fighter airplane are not shown in the ODS of the Theater TMO but these environment objects are described in the corresponding parts of the state descriptors for the command ship and the fighter airplane, respectively.

The Theater Space component in the ODS of the Theater TMO not only provides geographical information about the theater but also maintains the position information of every moving object in the theater. This information is used to determine the occurrences of collisions among objects and to recognize the departure of any object from the theater space to the outside.

In addition to the selection of sensors and actuators, the system engineer should decide on the deployment of the computer-based control system in the theater. The functions of the control computer system will be determined based on the control theory logic adopted. In this experimental development, we deployed two control computer systems; one inside the command post (in land) and the other in the command ship.

5.3 Step 2: Expansion of the Theater TMO into a TMO network

As the system engineer refines the single TMO representation of the theater, a component in the ODS of the Theater TMO may be taken out of the Theater TMO and form a new TMO. For example, the command post and the command ship can be separated out of the Theater TMO and become represented by separate TMO's, the Command Post TMO and the Command Ship TMO. When the new TMO's are created, the SvM's that serve as front-end interfaces of those new TMO's and the call links from the earlier born TMO's to the new objects should also be created. As a result, the Theater TMO becomes a network of three TMO's. The two new TMO's may describe or simulate the command post and the command ship more accurately than the Theater TMO in Figure 3 did.

The Command Ship TMO contains a control computer system and so does the Command Post TMO. The two control computer systems can be separated out of the Command Post TMO and the Command Ship TMO, respectively and become represented by two separate TMO's. This makes the theater to be represented by a network of five TMO's. By now, a requirement specification to be given to the computer engineer (team) has become ready. The full specification (in a form similar to Figure 4) of the network of six TMO's including the Alien TMO plus the following statement can form such a requirement specification:

> "Embed one control computer system in the Command Post and another in the Command Ship such that the computer systems control the chosen sensors (radar's) and the chosen interceptor launchers in order (1) to intercept incoming dangerous RV's by using air-borne launchers first, ground-based launchers next, and ship-borne launchers last, and (2) to move the command ship appropriately to further reduce the danger."

5.4 Step 3: Detailed design of the control computer system as an TMO network

Let us now consider only the control computer system to be housed in the command post in land. Initially, the computer engineer starts with a single TMO structuring of an abstract design of the control computer system. The ODS of the TMO contains several major data structures such as *Radar Data Queue* (RDQ). In the next step, the single TMO design is decomposed into multiple TMO's, each of which is centered around a major data structure contained in the ODS of the previous single TMO design. Figure 4 shows a partially detailed design specification of the RDQ TMO. Further details on this application scenario are referred to [Kim96d, Kim97c].

Radar_Data_Queue

Access Capability (to other TMO's)

- Flying Object Tracking Information (Receive_new_info_from_Radar_Data_Queue)

Object Data Store

Scan_Search_Result: MVD: x msec : list of {current_position, current_time}
Spot_Check_Predicted: MVD: y msec : list of {object_ID, predicted_position, predicted_time }
Spot_Check_Result: MVD: z msec : list of {current_position, current_time, predicted_time}

SpM

SpM1 *Send_New_Data_To_Flying_Object_Tracking_Information*

- For each spot-check result in Data Set [Spot_Check_Result], match with predicted status data in .....
  - Send the message to Flying_Object_Tracking_Information (via SvM request).
- For each scan search results in Data Set [Scan_Search_Result], construct a message .....
  - Send the message to Flying_Object_Tracking_Information (via SvM request).

AAC: **for** T = **from TMO_START** + WARMUP_DELAY_SECS **to TMO_START** + SYSTEM_LIFE_HOURS
every PERIOD **start-during** (T, T + START_WINDOW) **finish-by** T + DEADLINE

OutputSpec: <deadline: xxx msec> list of (object_ID, position, time)

SvM

SvM 1 < **Accept-via-Service_Request_Channel-with-Delay_Bound-of** ACCEPTANCE_DEADLINE
under MAX_REQUEST_RATE **finish-within** EXECUTION_TIME_LIMIT>

Receive_From_Radar_In_Land (pos_list)

- Receive from Radar_In_Land the information on all recent object detections.
- Update Data Set [Scan_Search_Result] or [Spot_Check_Result] according to the value of detection_type.

InitiationCond: Other SvM1 invocations are not in place.

InputSpec: pos_list = array of (return_type (=scan_search/spot_check), position, time, predicted_time)

SvM 2 < **Accept-via-** ..... >

Receive_From_Flying_Object_Tracking_Information (spot_list)

...

Figure 4 A partially detailed design specification of the RDQ TMO

Figure 5 depicts the fully decomposed network of TMO's in the CAMIN. The right-hand side of the network depicts the real-time simulator of the application environment, which consists of Radar TMO, Fighter-Airplane TMO, etc., whereas the left-hand side depicts the designs of the control computer systems. A prototype implementation of a home-grown timeliness-guaranteed OS kernel model called the DREAM kernel [Kim95b], was used in the experimental implementation of the CAMIN and its application environment simulator. This real-time distributed application software consists of nine different types of tailorable TMO's and runs on a network of three PC's. In this experimental effort, the TMO's were implemented in C++ with the support of a library named the DREAM library [Kim96a, Kim97d] which serves as a friendly API of the DREAM kernel implementation.

6. Advantages of the TMO based design and simulation

The TMO based uniform structuring approach brings the following major advantages which have been observed in our multiple experiments conducted so far:

(1) Strong traceability between requirement specification and design.

(2) Cost-effective high-coverage validation

(3) Autonomous subsystems and ease of maintenance

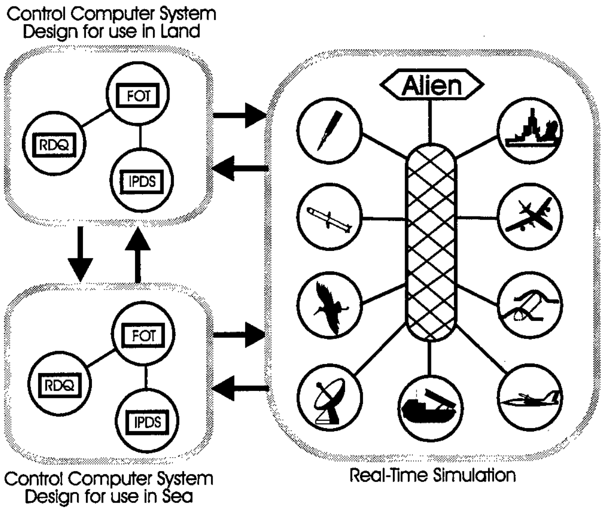


Figure 5. Network of TMO's in CAMIN
(Adapted from [Kim96c])

Prior to the development of the TMO structuring scheme, multiple experimental implementations of defense applications similar to the CAMIN had been conducted in the author's laboratory. Those early implementations were done by using conventional process structured design approaches without rigorous specification and analysis of timing aspects. Our experiences indicate that the TMO structured design approach yields at least several times improvement, if not an order-of-magnitude improvement, in design productivity over the conventional process structured design approaches.

7. Conclusion

The authors believe that the real-time object structuring technology and the real-time simulation technology are of vital importance in facilitating highly reliable engineering of large-scale RTCS's such as defense command-control systems, which the society has started demanding. These technologies will bring not only improved product reliability but also improved development economy. Considering that real-time application markets in the non-military world are about to explode due to the improved reliability of multimedia handling technologies, the needs of the real-time object structuring technology and the real-time simulation technology are as acute in the general data processing area as in the defense area. Therefore, although these technologies have not yet been established in mature forms, it is safe to bet that their advancement will be very rapid from now on.

As a sample of what is coming in the area of the real-time object structuring technology, the TMO technology was briefly presented in this paper. The uniform structuring of both RTCS's and real-time simulators of their application environments that is facilitated by the TMO scheme brings in significant improvements in various parts of the system engineering process, which in turn results in significantly improved design productivity. Although limited in scope, the RTCS development experiments conducted so far are strongly supportive of this positive assessment. However, to fully realize the potential, many new specification, design, and execution tools need to be developed.

Acknowledgment: The research work reported here was supported in part by US Navy, NSWC Dahlgren Division under Contract No. N60921-92-C-0204, in part by the University of California MICRO Program under Grant No. 96-169, in part by the California Transportation Department via the UCI Institute for Transportation Studies, in part by Hitachi, Ltd, in part by ETRI, in part by Postech, and in part by LG Electronics.

References

[Att91] Attoui, A. and Schneider, M., "An Object Oriented Model for Parallel and Reactive Systems", *Proc. IEEE CS 12th Real-Time Systems Symp.*, 1991, pp. 84-93.

[Boo91] Booch, G., *'Object-Oriented Design'*, Benjamin Cummings, CA, 1991.

[Dah72] Dahl, O.J., "Hierarchical Program Structuring", in Dahl, Dijkstra, & Hoare eds., *'Structured Programming'*, Aca. Press, NY, 1972.

[Ell90] Ellis, M.A. and Stroustrup, B., *'The Annotated C++ Reference Manual'*, Addison-Wesley Pub. Co., Reading, MA, 1990.

[Ish92] Ishikawa, Y., Tokuda, H., and Mercer, C. W. An Object-Oriented Real-Time Programming Language, *IEEE Computer* (Oct. 1992), 66-73.

[Kim94a] Kim, K.H. et al., "Distinguishing Features and Potential Roles of the TMO Model", *Proc. 1994 IEEE CS Workshop on Object-oriented Real-time Dependable Systems (WORDS)*, Oct. '94, Dana Point, pp.36-45.

[Kim94b] Kim, K.H. and Kopetz, H., "A Real-Time Object Model TMO.k and an Experimental Investigation of Its Potentials", *Proc. 1994 IEEE CS Computer Software and Applications Conf. (COMPSAC)*, Nov. 1994, Taipei, pp.392-402.

[Kim95a] Kim, K.H., "Toward New-Generation Real-Time Object-Oriented Computing", *Proc. IEEE CS 5th Workshop on Future Trends of Distributed Computing Systems (FTDCS)*, Cheju Island, Aug. '95, pp.520-529.

[Kim95b] Kim, K.H. et al., "A Timeliness-Guaranteed Kernel Model - DREAM Kernel and Implementation Techniques", *Proc. 1995 Int'l Workshop on Real-Time Computing Systems and Applications (RTCSA 95)*, Tokyo, Japan, Oct. 1995, pp.80-87.

[Kim96a] Kim, K.H. et al., "The DREAM Library Support for PCD and TMO.k programming in C++", *Proc. 1996 IEEE CS Workshop on Object-oriented Real-time Dependable Systems (WORDS)*, Feb. 96, Laguna Beach, pp. 59-68.

[Kim96b] Kim, K.H., Nguyen, C., and Park, C., "Real-Time Simulation Techniques Based on the TMO Modeling", *Proc. COMPSAC '96 (IEEE CS Software & Applications Conf.)*, Seoul, August 1996, pp.176-183.

[Kim96c] Kim, K.H., Kim, Y.S., and Kim, H.J., "An TMO Based Uniform Integrated Design of Real-Time Computing Systems and their Application Environment Simulators", *Proc. 2nd World Conf. on Integrated Design and Process Technology*, IDPT-Vol.2, Austin, TX, Dec. 1996, pp.106-113.

[Kim97a] Kim, K.H. and Subbaraman, C., "Fault-Tolerant Real-Time Objects", *Communications of the ACM*, January 1997, pp. 75-82.

[Kim97b] Kim, K.H., "Toward New-Generation Object-Oriented Real-Time Software and System Engineering" Invited paper, *SERI Journal*, Taejon, Korea, Vol.1, No.1, Jan. 1977, pp.1-23.

[Kim97c] Kim, K.H., "Uniform Object Structuring of Complex Systems and Environment Simulators", to appear in *Computer* (The IEEE CS Magazine), 1997.

[Kim97d] Kim, K.H., Subbara, C., and Bacellar, L., "Support for TMO Structured Programming in C++", to appear in *Control Engineering Practice* (an IFAC Journal), 1997.

[Kop90] Kopetz, H. and Kim, K.H., 1990, "Temporal Uncertainties in Interactions among Real-Time Objects", Proc. IEEE CS 9th Symp. on Reliable Distributed Systems, Huntsville, AL, Oct. 1990, pp.165-174.

[Rum91] Rumbaugh, J. et al., *'Object-Oriented Modeling and Design'*, Prentice Hall, NJ, 1991.

[Sel94] Selic, B., Gullekson, G., and Ward, P.T., *'Real-Time Object-Oriented Modeling'*, John Wiley & Sons, New York, 1994.

[Tak92] Takashio, K., and Tokoro, M., "DROL: An Object-Oriented Programming Language for Distributed Real-Time Systems", *Proc. OOPSLA*, 1992, pp. 276-294.

[Wor94] *'Proc. 1994 IEEE CS Workshop on Object-oriented Real-time Dependable Systems (WORDS), Oct. '94, Dana Point'*, IEEE CS Press, CA, 1995.

[Wor96] *'Proc. 1996 IEEE CS Workshop on Object-oriented Real-time Dependable Systems (WORDS), Feb. '96, Laguna Beach'*, IEEE CS Press, CA, 1996.

[Wor97] *'Proc. 1997 IEEE CS Workshop on Object-oriented Real-time Dependable Systems (WORDS), Feb. '97, Newport Beach'*, to appear in June 1997, IEEE CS Press, CA.

MorphoSys: An Integrated Re-configurable Architecture

Hartej Singh, Ming-Hau Lee, Guangming Lu,
Fadi J. Kurdahi, Nader Bagherzadeh, and Tomas Lang,

University of California, Irvine,
ET 544 F, Irvine, CA 92697, United States

Robert Heaton,
Obsidian Technology, and

Eliseu M. C. Filho,
Federal University of Rio de Janeiro, Brazil

Summary: In this paper, we present the MorphoSys re-configurable architecture, which combines a configurable array of processing elements with a RISC processor core. We provide a system-level model, describing the array architecture and the inter-connection network. We give several examples of applications that can be mapped to the MorphoSys architecture. We also show that MorphoSys achieves performance improvements of more than an order of magnitude as compared to other implementations and processors.

1. Introduction

Re-configurable computing systems are systems that combine programmable hardware with programmable processors. At one extreme of the computing spectrum, we have general-purpose processors that are programmed entirely through software. At the other extreme are application-specific ICs (Asics) that are custom designed for particular applications. The former has wider applicability, while the latter is specialized but very efficient. Re-configurable computing is a hybrid of the two approaches. It involves configuration or customization of hardware for a range of applications [4]. Conventionally, the most common devices used for re-configurable computing are field programmable gate arrays (FPGAs) [1]. FPGAs allow designers to manipulate gate-level devices such as flip-flops, memory and other logic gates. However, FPGAs have certain inherent disadvantages such as bit-level operation and inefficient performance for ordinary arithmetic or logic operations. Hence, many researchers have focused on a more general and higher level model of configurable computing systems. As a result, the PADDI [5], rDPA [6], DPGA [7], MATRIX [8], Garp [9], RaPiD [10,11], and Raw [12,13] are some of the systems that have been developed as prototypes of re-configurable computing systems. These are discussed briefly in a following section.

Target applications: Over the last decade, configurable computing systems have demonstrated significant potential for a range of applications. Many of these tasks (e.g. real-time signal processing) are computation-intensive and have high throughput requirements. Other applications are inherently complex (e.g. real-time speech recognition). In general, conventional microprocessor-based architectures fail to meet the performance needs for most of the applications in the realm of image processing, image understanding, signal processing, encryption, information-mining, etc. Automatic target recognition, feature extraction, surveillance, video compression are among those applications that have shown performance improvements of over an order of magnitude when implemented on configurable systems [4]. Other target applications for configurable systems are data parallel computations, convolution, stream processing, template matching, image filtering, etc.

Organization of paper: Section 2 provides definitions for terms relevant to re-configurable computing. Then, we present a brief review of previous research work in this sphere. Section 4 introduces the system model for MorphoSys, our prototype configurable computing system, MorphoSys. The following section (Section 5) describes the architecture of the basic cell of MorphoSys programmable hardware. Next, we discuss the mapping of a set of applications from image processing domains (video compression and automatic target recognition). We provide performance estimates for these applications and compare them with other systems and processors. Section 7 describes the MorphoSys simulation environment and graphical user interface. Finally, we present some conclusions from our research in Section 8.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

2. Taxonomy

In this section, we provide definitions for parameters that are typically used to characterize the design of a re-configurable computing system.

(a) *Granularity (fine versus coarse)*: This refers to the level of operation, i.e. bit-level versus word-level. Bit-level operations correspond to fine-grain granularity but coarse-grain granularity implies word-level operations. Depending upon the granularity, the configurable component may be a look-up table, a gate or an ALU-multiplier.

(b) *Depth of Programmability (single versus multiple)*: This is defined as the number of configuration planes resident in a re-configurable system. Some systems may have only a single resident configuration plane. This means that system functionality is limited to that plane. On the other hand, a system may have multiple configuration planes. In this case, different tasks may be performed by choosing varying planes for execution.

(c) *Re-configurability (static versus dynamic)*: A system may be frequently reconfigured for executing different applications. Re-configuration is either static (execution is interrupted) or dynamic (in parallel with execution). Single context systems can typically be reconfigured only statically. Multiple context systems favor dynamic reconfiguration

(d) Interface (remote versus local): A configurable system has remote interface if the system's host processor is not on the same chip/die as the programmable hardware. The system has a local interface if the host processor and programmable logic reside within the same chip.

(e) *Computation model*: For most configurable systems, the computation model may be described as either SIMD or MIMD. Some systems may follow the VLIW model.

3. Related Work

There has been considerable research effort to develop prototypes for configurable computing. In this section, we shall present the salient architectural features of each system.

The Splash [2] and DECPeRLe-1 [3] computers were among the first research efforts in configurable computing. Splash consists of a linear array of processing elements with limited routing resources. It is useful for linear systolic applications. DECPeRLe-1 is organized as a two-dimensional array of 16 FPGAs. The routing is more extensive, with each FPGA also having access to a column and row bus. Both systems are fine-grained, with remote interface, single configuration and static re-configurability.

Other research prototypes with fine-grain granularity include DPGA [7] and Garp [9]. Systems with coarse-grain granularity include PADDI [5], rDPA [6], MATRIX [8], RaPiD [10] and Raw [12].

PADDI [5] has a set of concurrently executing 16-bit functional units (EXUs). Each of these has an eight-word instruction memory. The communication network between EXUs uses crossbar switches for flexibility. Each EXU has dedicated hardware for fast arithmetic operations. Memory resources are distributed among the EXUs.

rDPA: The re-configurable data-path architecture (rDPA) [6] aims for better performance for word-level operations through data-paths wider than typical FPGA data-paths. The rDPA consists of a regular array of identical data-path units (DPUs). Each DPU consists of an ALU, a micro-programmable control and four registers. There are two levels of interconnection: local (mesh network of short wires) and global (long wires). The rDPA array is dynamically re-configurable.

MATRIX: This architecture [8] aims to unify resources for instruction storage and computation. The basic unit (BFU) can serve either as a memory or a computation unit. The 8-bit BFUs are organized in an array, where each BFU has a 256-word memory, ALU-multiply unit and reduction control logic. The interconnection network has a hierarchy of three levels (nearest neighbor, length four bypass connection and global lines).

RaPiD: This is a linear array of functional units [10], which is configured mostly to form a linear computation pipeline. The identical array cells each have an integer multiplier, three ALUs, six registers and three small local memories. A typical array has 8 to 32 of these cells. It uses segmented buses for efficient utilization of interconnection resources.

Raw: The main idea of this approach [12] is to implement a highly parallel architecture and fully expose low-level details of the hardware architecture to the compiler. The Re-configurable Architecture Workstation (Raw) is a set of replicated tiles, wherein each tile contains a simple RISC like processor, small amount of bit-level configurable logic and some memory for instructions and data. Each Raw tile has an associated programmable switch which connects the tiles in a wide-channel point-to-point interconnect.

DPGA: A fine-grain prototype system, the Dynamically Programmable Gate Arrays (DPGA) [7] use traditional 4-input lookup tables as the basic array element. Each cell can store 4 context words. DPGA supports rapid run-time reconfiguration. Small collections of array elements are grouped as sub-arrays that are tiled to form the entire array. A sub-array has complete row and column connectivity. Configurable crossbars are used for communication between sub-arrays.

Garp: This fine-grained approach [9] has been designed to fit into an ordinary processing environment, where a host processor manages main thread of control while only certain loops and subroutines use the re-configurable array for speedup in performance. The host processor is responsible for loading and execution of configurations on the re-configurable array. The instruction set of the host processor has been expanded to accommodate instructions for this purpose. The array is composed of rows of blocks. These blocks resemble CLBs of Xilinx 4000 series [25]. There are at least 24 columns of blocks, while number of rows implementation specific. The blocks operate on 2-bit data. There are vertical and horizontal block-to-block wires for data movement within the array. Separate memory buses move information (data as well as configuration) in and out of the array.

4. MorphoSys System Model

Figure 1 shows the organization of the MorphoSys re-configurable computing system. It is composed of a re-configurable array, a control processor, a data buffer and a DMA controller. It is coarse-grain (16-bit data-path), and the main thread of control is managed by an on-chip host processor. The programmable part is an 8 by 8 array of re-configurable cells (Figure 2), with multiple context words, operating in SIMD fashion. MorphoSys is targeted at image processing applications. Automatic target recognition and video compression (block motion estimation and discrete cosine transform) are some of the important tasks for which we have performed simulations. The system model and architecture details for the first implementation of MorphoSys (M1 chip) are described hereafter.

4.1 System Overview

Re-configurable Cell Array: The main component of MorphoSys is the Re-configurable Cell (RC) Array (Figure 2). It has 64 re-configurable cells, arranged as an 8 by 8 array. Each cell has an ALU/multiplier and register file (16-bit data-path). The RC Array functionality and interconnection network are configured through 32-bit context words. The context words are stored in a Context Memory in two blocks (one for rows and the other for columns). Each block has eight sets of sixteen contexts.

Host/Control processor: The controlling component of MorphoSys is a 32 bit RISC processor, called Tiny RISC. This is largely based on the design and implementation in [14]. Tiny RISC controls operation of the RC array, as well as data transfer to and from the array. Several new types of instructions were added to the Tiny RISC instruction set to enable it to perform these additional operations.

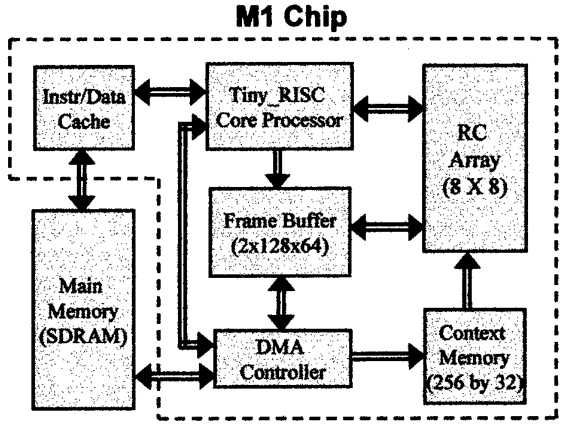


Figure 1: Block diagram of MorphoSys (M1 chip)

In addition to the RC Array and Tiny RISC processor, there is a Frame Buffer, designed primarily for storing image data. This buffer has two sets, each one sub-divided into two banks. The frame buffer is organized as a byte-addressable 256 word SRAM. Each word has eight bytes. The DMA Controller, provides an interface for data transfers between external memory (SDRAM) and the frame buffer or context memory of the RC array. It is essential to have an on-chip DMA Controller in order to efficiently process input and output data.

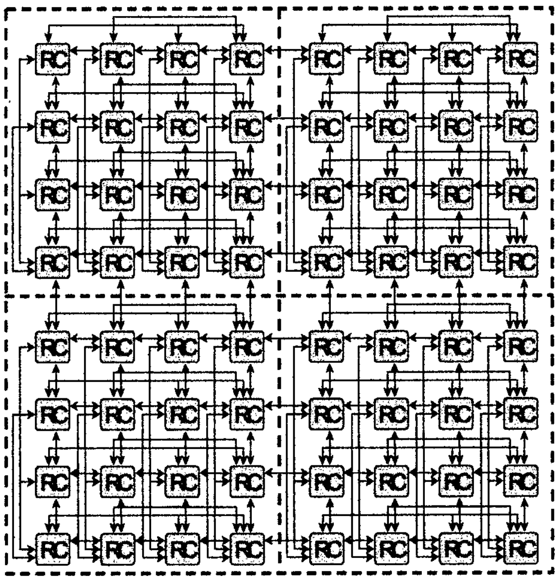


Figure 2: MorphoSys 8 x 8 Re-configurable Array

4.2 Program Flow

The MorphoSys system operates as follows: The Tiny RISC processor loads the configuration data from Main Memory into Context Memory through DMA Controller (Figure 1). Next, it enables the Frame Buffer to be loaded with image data from Main Memory. This data transfer is also done by the DMA unit. At this point, both configuration as well as data are ready. Now, Tiny RISC issues instructions to RC Array for execution. These instructions specify the particular context (among the multiple contexts in Context Memory) to be executed. Tiny RISC can also enable selective functioning of a row/column, and can access data from selected RC outputs.

4.3 Features of MorphoSys

The RC Array follows the *SPMD* (Single Program Multiple Data) model of computation. Each row/column is configured by one context, which serves as an instruction word. However, each of these cells operates on different data. This model serves the target applications (i.e. applications with large number of data-parallel operations) for MorphoSys very well.

In brief, the important features of the MorphoSys computation model are:

Coarse-level granularity: Each cell of the RC array function is configured by the context word. The context word specifies one of several instruction opcodes for the RC array, and provides control bits for input multiplexers. It also specifies constant values that are needed for computations.

Considerable depth of programmability: The context memory can store up to 16 contexts corresponding to a specific row and 16 contexts corresponding to a specific column. Our design provides the option of broadcasting contexts across rows or columns.

Dynamic reconfiguration capability: This is achieved by changing some portion of the context memory while the RC array is executing contexts from a different portion. For example, while the RC array is operating on the 16 contexts in row broadcast mode, the other 16 contexts for column broadcast mode can be reloaded. Context loads and reloads are done through Tiny RISC instructions.

Local Interface: The control processor (Tiny RISC) and the RC Array are on the same chip. This prevents I/O limitations from affecting performance. In addition, the memory interface is through an on-chip DMA Controller, for faster data transfers between external memory and the Frame Buffer. It also helps in decreasing the configuration loading time.

4.4 TinyRISC Instructions for MorphoSys

Several new instructions (Table 1) were introduced in the Tiny RISC instruction set for effectively controlling the MorphoSys RC Array operations. These instructions enable data transfer between main memory (SDRAM) and frame buffer, load configuration from main memory into context memory, and control RC array execution.

Table 1: Modified Tiny RISC Instructions for MorphoSys M1 chip

CBCAST	: Execute specific context in *RC Array*
DBCB	: Execute RC context, read two operand data into *RC Array* from *Frame Buffer*
LDCTXT	: Load context into *Context Memory*
LDFB	: Load data into *Frame Buffer* from memory
RCRISC	: Write data from *RC Array* to *Tiny RISC*
SBCB	: Execute RC context, read one operand data into *RC Array* from *Frame Buffer*
STFB	: Store data from *Frame Buffer* to memory
WFB	: Write data from specific column of *RC Array* into *Frame Buffer*

There are two categories of these instructions: DMA instructions and RC instructions. The DMA instruction fields specify load/store, memory address (indirect), number of bytes/contexts to be transferred and frame buffer or context memory address. The RC instruction fields specify address of context to be executed, address of frame buffer (if RC needs to read/write data) and broadcast mode (row/column). The instructions are summarized in Table 1.

5. RC Array Architecture

In this section, we describe three major features of MorphoSys. First, the architecture of each re-configurable cell is detailed (Figure 3), with description of different functional, storage and control components. Next, we discuss the context memory, its organization, field specification and broadcast mechanism. Finally, we describe the three-level hierarchical interconnection network of the RC array.

5.1 Re-configurable Cell Architecture

The re-configurable cell (RC) array is the programmable core of MorphoSys. It consists of 64 identical Re-configurable Cells (RC) arranged in a regular fashion to form an 8x8 array (Figure 2). The basic configurable unit, is the RC (Figure 3). Its functional model is similar to the data-path of a conventional processor, but the control is modeled after the configuration bits in FPGA chips. As Figure 3 shows, the RC comprises an ALU-multiplier, a shift unit, and two multiplexers for ALU inputs. There are registers at the output and for feedback, and a register file with four registers. A context word, loaded from Context Memory and stored in the context register (Section 5.2), defines the functionality of the ALU and direction/amount of shift at the output. It provides control bits to input multiplexers and determines which registers are written after an operation. In addition, the context word (stored in the context register) can also specify an immediate value (referred to as a constant).

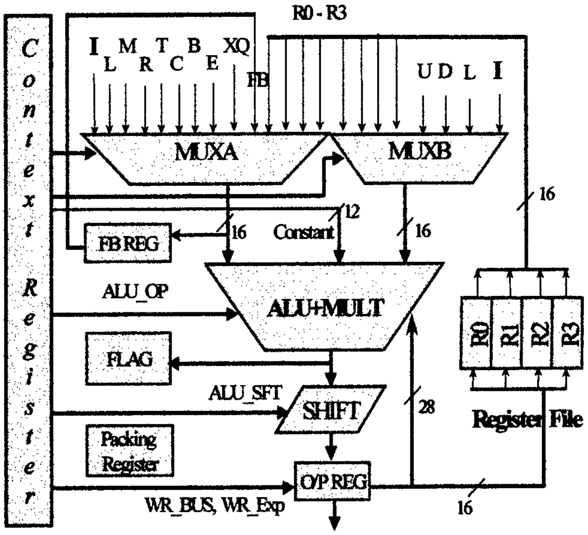


Figure 3 : Re-configurable Cell Architecture

ALU-Multiplier unit: The ALU has 16-bit inputs, and the multiplier has 16 by 12 bit inputs, producing an output of up to 28 bits. Externally, the ALU-multiplier has four input ports. Two ports, Port A and Port B are for data from outputs of input multiplexers. The third input (12 bits) takes a value from the constant field in the context register (Figure 4). The fourth port takes its input from the output register. The ALU has standard logic functions. Among its arithmetic functions are addition, subtraction and a function to compute absolute value of difference of two numbers. The ALU also has some functions that take one operand from port A, and the other from constant input port. The unit is capable of doing a multiply-accumulate operation in one cycle, wherein two data are multiplied and added to the previous output value. The ALU adder has been designed for 28 bit inputs. This prevents loss of precision during multiply-accumulate operation, even though each multiplier output may be much more than 16 bits, i.e. a maximum of 28 bits.

Input multiplexers: The two input multiplexers select one of several inputs for the ALU. Mux A is a 16-to-1 mux, whereas Mux B is an 8-to-1 mux (Figure 3). Mux A provides inputs from the four nearest neighbors, and from the other cells in the same row and column within the quadrant. It also provides an express lane input (as explained in sub-section on Interconnection network), array data bus input, a feedback input, a cross-quadrant input and four inputs for register file. Mux B provides four register file outputs, array data bus input and inputs from three of the nearest neighbors.

Registers: The register file is composed of four registers (16-bit), which prove adequate for most applications. The output register is 32 bits wide (to accommodate intermediate results of multiply-accumulate instructions). The shift unit is also 32 bits wide and can perform logical right or left shifts of 1 to 15 bits (Figure 3). A flag register indicates sign of input operand at port A of ALU. It is zero for positive operands and one for negative operands. A flag is useful when the operation to be performed depends upon the sign of the operand, as in the quantization step during image compression. A feedback register is also available in case an operand needs to be re-used, as in motion estimation.

Custom hardware: This is used to implement special functions, especially bit-level function, for e.g. a one's counter or a packing register (for merging binary data into words).

5.2 Context Memory

The configuration information for the RC Array is stored in the Context Memory. Also, each RC has a Context Register, in which the current context word is stored. The Context Memory is organized into two blocks with each block having eight sets of sixteen contexts.

Context register: This register (32 bits), specifies the context (i.e. the configuration control word) for the RC. The Context Register is a part of each re-configurable cell, whereas the Context Memory (SRAM) is separate from the RC Array (Figure 1).

The different fields for the context word are defined in Figure 4. The field ALU_OP specifies ALU function. The control bits for Mux A and Mux B are specified in the fields MUX_A and MUX_B. Other fields determine the registers to which the result of an operation is written (REG #), and the direction (RS_LS) and amount of shift (ALU_SFT) applied to output. One interesting feature is that the context includes a 12 bit-field for the constant. If the ALU-Multiplier functions do not need a constant, an ALU-Multiplier sub-operation is defined. These sub-operations are used to expand the functionality of the ALU unit.

The context word also specifies whether a particular RC writes to its row/column express lane (WR_Exp). Whether or not the RC array will write out the result to

Frame Buffer, is also specified by the context data (WR_BUS). (Figure 4)

The programmability of interconnection network is also derived from the context word. Depending upon the context, an RC can access the input of any other RC in its column or row within the same quadrant, or else select an input from its own register file. Functional programmability is achieved by configuring the row/column ALUs to perform functions as specified in the context word.

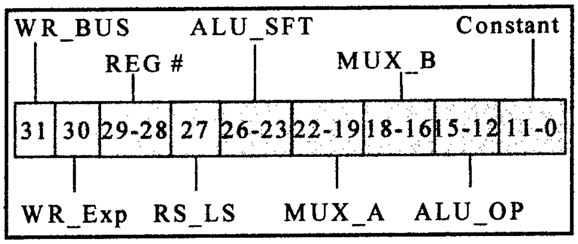


Figure 4: RC Context word definition

Context broadcast: The major focus of the RC array is on data-parallel applications, which exhibit a definite regularity. Based on this idea of regularity and parallelism, the context is broadcast to a row (column) of RCs. That implies that all eight RCs in a row (column) share the same context, and perform the same operations. For example, for DCT computation, eight 1-D DCTs need to be computed, across eight rows. This is easy to achieve with just eight context words to program eight rows of 64 RCs.

Context memory organization: The context memory is designed as an SRAM, with total memory of 256 (16X16) 32 bit words (Figure 1). The context may be broadcast across rows or columns. Corresponding to that, sixteen context sets/cells are needed (eight for row broadcast to eight rows, and eight for column broadcast to eight columns). The computation model for the RC supports multiple contexts. In other words, there are multiple context memory words for each of the above sixteen sets. Based on studies of relevant applications, a depth of sixteen for each context set has been found sufficient for most applications studied for this project. Each context word is 32 bits, there are sixteen words in a context set, and there are sixteen context sets, giving the above figures.

Dynamic reconfiguration: MorphoSys supports dynamic re-configuration. This implies that while the RC Array is executing a series of context words, another set of contexts may be reloaded from main memory through the DMA controller. The context depth allows the RC Array to operate continuously even when portions of the Context Memory need to be changed.

5.3 Interconnection Network

The RC interconnection network is comprised of three hierarchical levels.

RC array mesh: The underlying network throughout the array is a 2-D mesh. This provides nearest neighbor connectivity. Each RC is connected to its North, South, East and West neighbors (Figure 5).

Intra-quadrant (complete row/col) connectivity: The second layer of connectivity is at the quadrant level (a quadrant is a 4 by 4 RC group). In the current MorphoSys specification, the RC array has four quadrants (Figure 2). Within each quadrant, each cell has complete connectivity in two dimensions, as shown in Figure 5. Each cell can access the output of any other cell in its row (column).

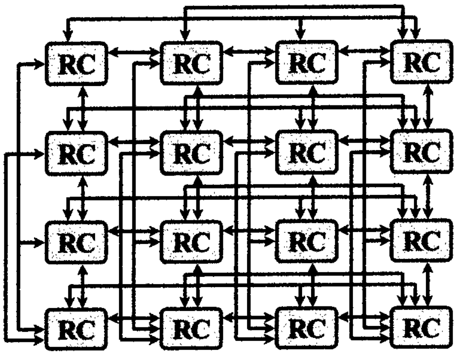


Figure 5: Connectivity within a quadrant

Inter-quadrant (express lane) connectivity: At the highest or global level, there are buses for routing connections between adjacent quadrants. These buses, also called express lanes, run across rows as well as columns. Figure 6 shows two express lanes going in each direction across a row. These lanes can supply data from any one cell (out of four) in a row (column) of a quadrant to other cells in adjacent quadrant but in same row (column). This means that up to four cells in a row (column) may access the output value of any one of four cells in the same row (column), but in an adjacent quadrant. The express lanes greatly enhance global connectivity. Even irregular communication patterns, that require extensive interconnections, can be handled efficiently. For e.g., an eight-point butterfly is accomplished in only three cycles.

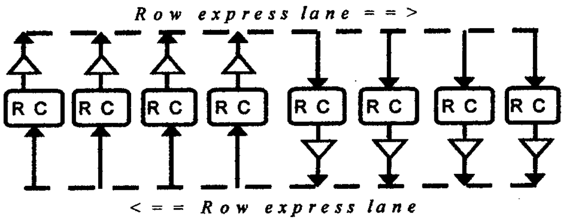


Figure 6: Express lane connectivity (between two groups of cells in same row but adjacent quadrants)

Data bus: A 128-bit data bus from Frame buffer to RC array is linked to column elements of the array. It provides two operands of eight bits to each of the eight column cells. It is possible to load two operand data (port A and port B) in an entire column in one cycle. Thus, only eight cycles are required to load the entire RC array. The outputs of RC elements of each column are written back to frame buffer through Port A data bus.

Context bus: When a Tiny RISC instruction specifies that a particular group of context (configuration word) be executed, these contexts must be distributed to each RC from the Context Memory. The context bus communicates this context data to the Context Register in each RC in a row (column). Each context word is 32 bits wide, and there are eight rows (columns), hence the context bus is 256 bits wide. When the row (column) contexts are activated, the same context word is broadcast to all RCs in a particular row (column).

6. MorphoSys: Application Mapping

In this section, we shall consider the mapping of several relevant applications to the MorphoSys architecture. These applications were chosen because of their computation-intensive nature. Among the video compression applications are motion estimation, DCT and quantization. Automatic target recognition is an important military application. We also provide performance estimates (based on a first order analysis) for each application.

6.1 Example: Motion Estimation

Motion estimation is widely adopted in video compression to identify redundancy between frames. The most popular technique for motion estimation is the block-matching algorithm because of its simple hardware implementation [15]. Block matching algorithm is also recommended by several standard committees (e.g., MPEG and H.261 standards) [16]. Among several possible searching methods, the expense of full search block matching (FSBM) is obviously greatest. FSBM, however, gives the optimal solution and low control overhead and is found an ideal candidate for VLSI implementation.

Typically, the mean absolute difference (MAD) criterion is the one implemented in VLSI due to its relative accuracy and compactness of implementation in hardware [16]. With the MAD criterion, the FSBM can be formulated as follows:

$$MAD(m, n) = \sum_{i=1}^{N} \sum_{j=1}^{N} | R(i, j) - S(i+m, j+n) | ,$$

$$-p <= m, \; n <= q$$

where p and q are the maximum displacements, $R(i, j)$ is the reference block of size N x N pixels at coordinates (i, j), and $S(i+m, j+n)$ is the candidate block within a search area of size $(N+p+q)^2$ pixels in the previous frame. The displacement vector is represented by (m, n), and the motion vector is determined by the least $MAD(m, n)$ among all the $(p+q+1)^2$ possible displacements within the search area. Table 2 shows the specification of a video codec.

Table 2 : Specification of a Video Codec

Image Size	352 x 240	Pixels
Frame Rate	15	Frames/s
Block size	8 x 8	Pixels
Max. Displacement	$-8 \le m, n \le 8$	Pixels

Figure 7 shows the configuration of RC array for FSBM computation. The operations of RC array are denoted row by row. Initially, one reference block and the search area associated with it are loaded into one set of the frame buffer, and then the RC array starts the matching process for the reference block resident in the frame buffer. During the computation, another reference block and the search area associated with it are loaded into the other set of the frame buffer so that the loading and computing time are overlapped.

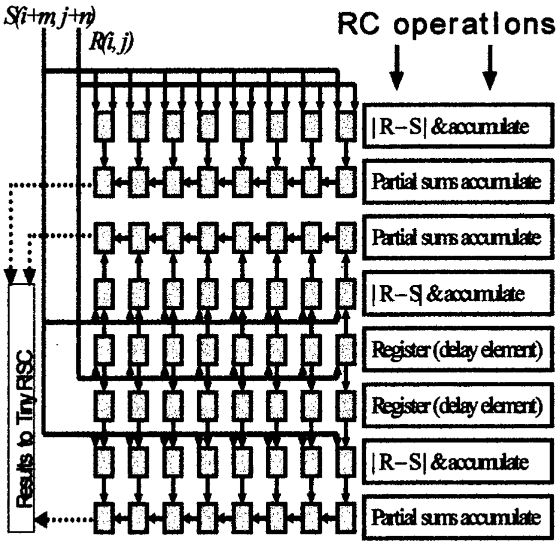


Figure 7: Configuration of RC Array for Full Search Block Matching

For each reference block, three consecutive candidate blocks are matched concurrently in the RC array. As shown in Figure 7, each RC in first, fourth, and seventh row performs $P_j = \Sigma_{1 \le i \le 8} \mid R(i, j) - S(i+m, j+n) \mid$. The data of the reference block is sent to the first row and passed to the fourth row and seventh row through delay elements. The eight partial sums (P_j) generated by the first, fourth, and seventh row are then passed to the second, third, and eighth row respectively to perform $MAD(m, n) = \Sigma_{1\leq j\leq 8} P_j$. Subsequently, three MAD values corresponding to three candidate blocks are sent to Tiny RISC for comparison, and the RC array starts the block matching of another three candidate blocks.

Computation cost: Based on the computation model shown above, it takes ten clock cycles to finish the matching of three candidate blocks. There are $(8+8+1)^2 = 289$ candidate blocks in each search area, and it takes a total of (102x10) 1020 cycles to finish the matching of the whole search area. According to the specification of Table 2, the image size is 352x240 pixels which consists of 44x30 = 1320 reference blocks. The total processing time of an image frame is 1320x1020 = 1346400 cycles. The anticipated clock rate of MorphoSys is 100 MHz, so the computation time would be $1346400 \times 10 \times 10^{-9} \approx 13.5$ ms. This is much smaller than frame period of 1/15 s shown in Table 2.

Performance Analysis: MorphoSys performance is compared with the ASIC architecture implemented in [15] and Intel MMX instructions [24] using the criterion of the number of cycles needed for matching one 8x8 reference block against its search area of 8 pixels displacement. The result is shown in Table 3.

Table 3: Performance Comparison

System	MorphoSys (8X8)	ASIC [15]	MMX [24]
(cycles)	1020	581	28900

The processing cycles of MorphoSys are about two times compared to ASIC design in [15], however, MorphoSys provides the advantage of hardware reuse. The number of cycles needed for MMX is about 28 times more than the cycles needed for MorphoSys, which shows the greater effective computing power of MorphoSys.

6.2 Example: Discrete Cosine Transform

The Discrete cosine transform (DCT) [17] and inverse discrete cosine transform (IDCT) form an integral part of the JPEG [19] and MPEG [20] standards. MPEG encoders use both DCT and IDCT, whereas IDCT is used in MPEG decoders. Compression is achieved in MPEG through the combination of DCT and quantization.

Fast algorithms for DCT: In the following analysis, we consider one of several fast DCT algorithms. This algorithm [18] for a fast 8-point 1-D DCT involves 16 multiplications and 26 additions, leading to 256 multiplications and 416 additions for a 2-D implementation. The 1-D algorithm is first applied to rows (columns) of input 8x8 image block, and then to columns (rows). The eight row (column) DCTs may be computed in parallel, for high throughput.

Mapping to RC Array: The block size for DCT in most image and video compression standards is 8x8. The RC array also has a size of 8x8. Thus, each pixel of the image block may be directly mapped to each RC. The input block is loaded into the array, such that one pixel value is stored in each RC. The next step is to perform eight 1-D DCTs along the rows (columns). In other words, row (column) parallel operations are required. The context for configuration needs to be broadcast along columns (rows). Under this condition, different RCs within a row (column) of the array communicate using three-layer interconnection network to compute outputs for 1-D DCT. The coefficients needed for the computation are provided as constants in context words. When 1-D DCT along rows (columns) is complete, the next step is to compute 1-D DCT along columns (rows). For column parallel operation, context words are broadcast along rows. Once again, RCs within a column communicate using the three-layer interconnections, to compute the results. As previously, coefficients are provided through context words. At the conclusion of this step, the 2-D DCT is complete.

Some points may be noted: first, all rows (columns) perform the same computations, hence they can be configured by a common context (thus enabling broadcast of context word). Second, the RC array provides the option of broadcasting context either across rows or across columns. This allows computation of second 1-D DCT without transposing the data. Elimination of the transpose operation saves a considerable amount of cycles, and is important for high performance.

Sequence of steps:

Loading the input data: The first step is to load the input image pixel block into the RC array. This data block is previously loaded into the frame buffer from external memory. The data bus between the frame buffer and the RC array is wide enough to load eight pixels at a time. The entire block is loaded in eight cycles.

Row-column approach: Following the separability property, the 2-D DCT is carried out in two steps. First, a 1-D DCT along rows is computed. Next, we compute 1-D DCT on the results along columns (Figure 8). Each sequence of 1-D DCT [18] involves:

i. *Butterfly computation:* Once the data is available, the next step is to compute the intermediate variables. The inter-quad connectivity layer of express lanes enables completion in three cycles.

ii. *Computation and re-arrangement*: In the next few steps, the actual computations pertaining to 1-D DCT algorithm [2] are performed. For 1-D

DCT (row/column), the computation takes six cycles. An extra cycle is used for re-arrangement of computed results.

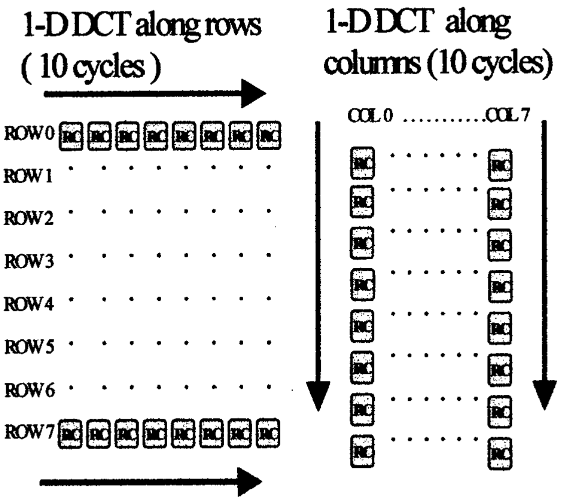


Figure 8: Computation of 2-D DCT across rows/columns (without transposing)

Computation cost: The cost for computing 2-D DCT on an 8x8 block of the image is as follows: 8 cycles for data input, 6 cycles for butterfly, and 12 cycles for both 1-D DCT computations. 2 cycles are used for re-arrangement of data. For 1024 by 768 pixel image, it would take 3.44 ms @ 100 MHz to complete the 2-D DCT computation.

Performance analysis: MorphoSys requires 21 cycles to complete 2-D DCT on 8x8 block of pixel data. This is in contrast to 240 cycles required by Pentium MMX ™ [23] . Even this number is achieved by using special MMX instructions in Pentium instruction set, assuming that all data is present in the cache. MorphoSys takes 8 cycles to load data and 8 for write-back while Pentium cache miss penalty is 12 cycles. Notably, MorphoSys performance scales linearly with the array size. For a 256 element RC array, the throughput for 2-D DCT algorithm would increase four times, giving an effective performance of 5 cycles per 8x8 block. The performance figures are summed up in Table 4.

Table 4: DCT Performance Comparison (in cycles)

MorphoSys (8 X 8)	MorphoSys 2 (8 X 8)	MorphoSys 4 (8X 8)	Pentium MMX™
21	**10**	**5**	**240**

6.3 Example: Automatic Target Recognition (ATR)

Automatic Target Recognition (ATR) is the machine function of detecting, classifying, recognizing, and identifying an object without the need of human intervention. ATR is one of the most computation intensive applications in existence. The ACS Surveillance Challenge has been quantified as the ability to search 40,000 square nautical miles per day with a resolution of one meter [21]. The computation levels for this problem when the targets are partially obscured reaches the hundreds-of-teraflops range. By considering the number of computations that must be carried out for real-time images, it is obvious that a hardware, which provides great computing power, is essential for solving ATR problem. In the previous sections, we have shown that MorphoSys with 8x8 RC array can perform FSBM motion estimation on a 352x240 pixels image at 70 frames per second. We believe that MorphoSys has a great opportunity to provide the computing power for ATR.

Currently, there are many algorithmic choices available to implement an ATR system [22]. A simplified ATR processing model used in this paper is shown in Figure 9 [23]. We assume that Synthetic Aperture Radar (SAR) images generated by the radar imager in real time are used. SAR images consist of 8-bits pixels and are input to a focus-of attention processor to identify the regions of interest (ROI). These ROI's are thresholded to generated binary images and the binary images are then matched against binary target templates.

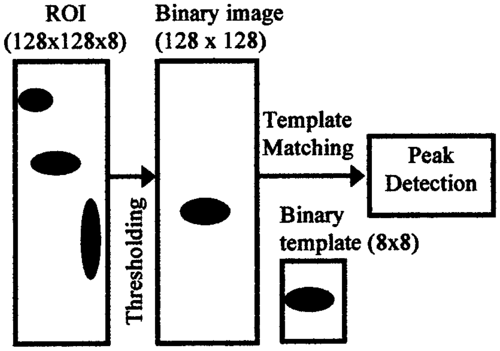


Figure 9: A Simplified ATR Processing Model

Before the thresholding, 64 pixels of ROI are loaded into RC array. Each pixel is then compared with the threshold value T and is set to a binary data based on the following equation:

If $A_{ij} - T > 0, \quad A_{ij} \leftarrow 1$
If $A_{ij} - T < 0, \quad A_{ij} \leftarrow 0,$ *where* A_{ij} *represents the 8-bits pixels in ROI*

The threshold value is assumed to be a pre-selected constant and is put in the context at compile time. Because 2's complement representation is used, the most significant bit of the output register represents the result of the thresholding. There is an 8-bits packing register in each RC of the first column. These registers are used to collect the thresholding results of the RCs in the same row. The data in the packing registers are then sent back to the frame buffer, and another 64 pixels of ROI are loaded to RC array for thresholding.

After the thresholding, a 128x128 binary image is generated and is stored in the frame buffer. This binary image is then matching against the target template. Each target consists of 72 templates at five degree rotations. Each row of the 8x8 target template is packed as an 8-bits number and eight templates are resident in the RC array concurrently. The template matching is similar to the FSBM described in the previous section. All of the candidate blocks in the ROI are correlated with the target template. One column of the RC array performs matching of one target template and eight target templates can be matched concurrently in the 8x8 RC array. Figure 10 illustrates the matching process in each RC.

In order to perform bit-level template matching, two bytes (16 bits) of image data are input to a RC. In the first step, the 8 most significant bits of the image data are ANDed with the template data and a special adder tree is used to count the number of 1's of the ANDed output. The result is passed to the peak detector. Then, the image data is shifted left one bit and the process is repeated again to perform the matching of the second block. After the image data is shifted eight times, new 16-bit data is loaded and the RC Array starts another matching of eight consecutive candidate blocks.

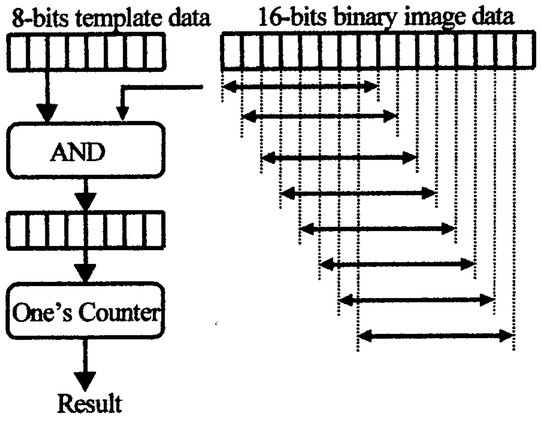


Figure 10: Matching Process in Each RC

Performance analysis: For performance analysis, we choose the system parameters that are implemented in [23]. The image size of ROI is 128x128 pixels, and the template size is 8x8 bits. For 16 pairs of target templates, the processing time is approximately 16 ms in MorphoSys assuming a clock of 100 MHz. This is more than one order of magnitude less than the 210 ms processing time for the system implemented in [23], which uses Xilinx XC4010 FPGAs. The performance figures are listed in Table 5.

Table 5: ATR Performance Comparison (ms) (MorphoSys @ 100 MHz)

MorphoSys (8 X 8)	MorphoSys 2 (8 X 8)	MorphoSys 4 (8X 8)	Xilinx FPGA
16	**8**	**4**	**210**

7. Interactive Software Environment

The MorphoSys re-configurable system has been specified in behavioral VHDL. We have coded a simple assembler-like parser, *mload*, for context generation. This context is then used as input for simulation. We have simulated motion estimation algorithm, the discrete cosine transform, and automatic target recognition among others.

A graphical user interface, *mView*, has been prepared for programming and observing simulation behavior for MorphoSys. This interface is based on Tcl/Tk [26]. It operates in one of two modes: programming mode or simulation mode. In the programming mode, the user sets functions and interconnections corresponding to each context (row/column broadcasting) for the application. Then, *mView* generates a context file for representing the user-specified application.

In the simulation mode, *mView* takes as input either a context file, or a simulation output file. For either of these, it provides a graphical display of the interconnections and RC outputs at each cycle of the simulation. Use of *mView* has been found to decrease the programming time, and also increase debugging efficiency.

8. Conclusions

In this paper, we presented a new model of re-configurable architecture in the form of MorphoSys, and mapped several applications to it. The results have shown that this architectural model enables impressive performance for several target applications. We plan to implement MorphoSys on an actual chip for practical laboratory evaluation. We conceive that MorphoSys may be the precursor of a generation of general-purpose processors that have a specialized re-configurable component, designed for image and signal processing applications.

9. Acknowledgements

This research is supported by Defense and Advanced Research Projects Agency (DARPA) of the Department of Defense under contract number F-33615-97-C-1126. We express thanks to Prof. Walid Najjar for his incisive comments. We acknowledge the contributions of Maneesha Bhate, Matt Campbell, Benjamin U-Tee Cheah, Alexander Gascoigne, Nambao Van Le, Robert Powell, Rei Shu, Lingling Sun, Cesar Talledo, Eric Tan, Tom Truong, and Tim Truong; all of whom have been associated with the development of MorphoSys.

References:

1. Stephen Brown and J. Rose, "Architecture of FPGAs and CPLDs: A Tutorial," *IEEE Design and Test of Computers*, Vol. 13, No. 2, pp. 42-57, 1996
2. M. Gokhale et al, "Building and Using a Highly Parallel Programmable Logic Array," *IEEE Computer*, pp. 81-89, Jan. 1991
3. P. Bertin, D. Roncin, and J. Vuillemin, "Introduction to Programmable Active Memories," in *Systolic Array Processors*, J. McCanny, J. McWhirther and E. Swartslander, eds., Prentice Hall, NJ, 1989, pp. 300-309
4. W. H. Mangione-Smith et al, "Seeking Solutions in Configurable Computing," *IEEE Computer*, pp. 38-43, December 1997
5. D. C. Chen and J. M. Rabaey, "A Re-configurable Multi-processor IC for Rapid Prototyping of Algorithmic-Specific High-Speed Datapaths," *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 12, December 1992
6. R. Hartenstein and R. Kress, "A Datapath Synthesis System for the Re-configurable Datapath Architecture," *Proc. of Asia and South Pacific Design Automation Conference*, 1995, pp. 479-484
7. E. Tau, D. Chen, I. Eslick, J. Brown and A. DeHon, "A First Generation DPGA Implementation," *FPD'95, Canadian Workshop of Field-Programmable Devices*, May 29–Jun1, 1995
8. E. Mirsky and A. DeHon, "MATRIX: A Re-configurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources," *Proc. IEEE Sym. on FPGAs for Custom Comp. Mach.*, 1996, pp.157-66
9. J. R. Hauser and J. Wawrzynek, "Garp: A MIPS Processor with a Re-configurable Co-processor," *Proc. of the IEEE Symposium on FPGAs for Custom Computing Machines*, 1997
10. C. Ebeling, D. Cronquist, and P. Franklin, "Configurable Computing: The Catalyst for High-Performance Architectures," *Proc. of IEEE Int'l Conference on Application-specific Systems, Architectures and Proc.*, July 1997, pp. 364-72
11. C. Ebeling, D. Cronquist, P. Franklin, J. Secosky, and S. G. Berg, "Mapping Applications to the RaPiD configurable Architecture," *Proc. IEEE Symposium of Field-Programmable Custom Computing Machines*, Apr 1997, pp. 106-15
12. E. Waingold et al, "Baring it all to Software: The Raw Machine," *IEEE Computer*, Sep 1997, pp. 86-93
13. E. Waingold et al, "The RAW Benchmark Suite: computation structures for general-purpose computing," *Proc. IEEE Symposium on Field-Programmable Custom Computing Machines*, FCCM 97, 1997, pp. 134-43
14. A. Abnous, C. Christensen, J. Gray, J. Lenell, A. Naylor and N. Bagherzadeh, " Design and Implementation of the Tiny RISC microprocessor," *Microprocessors and Microsystems*, Vol. 16, No. 4, pp. 187-94, 1992
15. C. Hsieh and T. Lin, " VLSI Architecture For Block-Matching Motion Estimation Algorithm," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 2, pp. 169-175, June 1992.
16. A. Bugeja and W. Yang, " A Re-configurable VLSI Coprocessing System for the Block Matching Algorithm," *IEEE Trans. On VLSI Systems*, vol. 5, September 1997.
17. N. Ahmed, T. Natarajan, and K.R. Rao, "Discrete cosine transform," *IEEE Trans. On Computers*, vol. C-23, pp. 90-93, Jan 1974
18. W-H Chen, C. H. Smith and S. C. Fralick, "A Fast Computational Algorithm for the Discrete Cosine Transform," *IEEE Trans. on Comm.*, vol. COM-25, No. 9, September 1977
19. *ISO/IEC JTC1 CD 10918*. Digital compression and coding of continuous-tone still images – part 1, requirements and guidelines, ISO, 1993 (JPEG standard)
20. *ISO/IEC JTC1 CD 13818*. Generic coding of moving pictures and associated audio: video, ISO, 1994 (MPEG-2 standard)
21. *Challenges for Adaptive Computing Systems*, Defense and Advanced Research Projects Agency, at www.darpa.mil/ito/research/acs/challenges.html
22. J. A. Ratches, C. P. Walters, R. G. Buser, and B. D. Guenther, " Aided and Automatic Target Recognition Based Upon Sensory Inputs From Image Forming Systems," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, September 1997.
23. J. Villasenor, B. Schoner, K. Chia, C. Zapata, H. J. Kim, C. Jones, S. Lansing, and B. Mangione-Smith, " Configurable Computing Solutions for Automatic Target Recognition," *Proceedings of IEEE Conference on Field Configurable Computing Machines*, April 1996.
24. *Intel Application Notes for Pentium MMX*, http://developer.intel.com/
25. *Xilinx, the Programmable Logic Data Book*, 1994
26. *Practical Programming in Tcl and Tk*, 2nd edition, by Brent B. Welch, Prentice-Hall, 1997

Using Genetics-based algorithms for Mission Systems applications

A Krouwel & C Williams
Strategic Communications and Networks Dept.
Satellite Communications Centre
DERA Defford
Worcestershire WR8 9DU
UK

1. SUMMARY

New methods which can deal with Mission Systems problems of the sheer magnitude now encountered, and can continue to perform under conditions of extreme noise and uncertainty, must be developed in order to take full advantage of the latest technological developments.

Genetics-based algorithms (GBAs) are a powerful search technique, that cope with the complex, deceptive and noisy problems that traditional techniques have problems with.

They supply a range of multiple, diverse solutions to a problem with high levels of robustness in dealing with unexpected or noisy situations.

GBAs are suitable for problem domains including:
a) planning and scheduling;
b) developing optimal strategies;
c) autonomous agents within simulations;
d) robust, adaptive, autonomous systems.

The approach is not perfect. The fact that Genetics-Based Algorithms do not take into account any domain knowledge, can lead to performance degradation, when compared to *problem specific* techniques. However, many other techniques can *only* be used (or used effectively) if the problem domain can be sufficiently analysed, and understood.

GBAs also do not tend to suffer from an explosion of computational time required, as problems are scaled up, in the same way that many other techniques do.

The use of co-evolution develops robust, *general* solutions which deal with new or changing conditions. This is appropriate to many military applications (as there is almost always an opponents view to be considered).

It would appear, that one way or another Genetics-Based Algorithms, will be part of, or affect the development of, many future military systems.

2. INTRODUCTION

2.1 Background

The military must constantly keep track of, and exploit, state of the art technology in order to maintain its position relative to potential threats.

Battlefield systems need to:
a) react more quickly;
b) take account of more information;
c) achieve better accuracy (in the face of increasing noise, ranges and high-tech counter-measures);
d) have more 'built-in' intelligence.

Military commanders require more and more support in order to be able to take advantage of the available data, and plan under increasingly complex constraints and conditions. Consequently, the techniques required to provide these systems must improve at the same rate.

Traditional approaches could be relied on to give support for the more clear cut 'straightforward', lower level tasks, leaving the commanders to use their *experience* and *common-sense* to make the more complex decisions, and handle the problems associated with the 'fog-of-war'.

As the requirement for decision support for the larger, more complex, tasks has increased, it has become clear that a technological leap forward in the methods used to provide this support is urgently needed.

New methods which can deal with problems of the sheer magnitude now encountered, and can continue to perform under conditions of extreme noise and uncertainty must be developed.

One class of techniques which may fulfil this role are Genetics-based Algorithms[1]. The potential for exploiting these techniques within future Mission Systems Applications is presented in this paper.

Despite the wide range of applications falling within the Mission Systems domain, most can be regarded as (complex) optimisation problems of one kind or another.

Genetics-based algorithms (GBA's) have been applied to many types of optimisation tasks, ever since they were first proposed in the 1970's, and have shown the potential to handle complex problems, and noisy environments. GBAs

[1] The term Genetics-based algorithm is used within this paper to include Genetic Algorithms (GA), Genetic Programming (GP), Evolutionary Strategies, and any other techniques which stem from a study of the biological and mathematical processes involved in natural evolution.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

should therefore have a lot to offer to Mission Systems developers.

These types of algorithms are currently being developed and evaluated against, a wide variety of military applications by various departments within DERA[2]. Studies performed to date have confirmed the potential of these techniques to perform a vital role in future military systems.

2.2 Scope

This paper will attempt to describe some typical Mission Systems problems and how they might benefit from the use of GBAs. In particular it will outline some of the projects which are currently on-going within DERA, where the possible exploitation of this technology is being investigated.

3. RELEVANT APPLICATION TYPES

3.1 Planning systems

This can be split into strategic planning problems (i.e. long-term, off-line planning) and tactical planning (i.e. more reactive responses or on-line systems).

Most of the applications described in detail in this paper either fall directly into the category of planning systems:
or involve some elements of planning such as the convoy movement problem.

Much research into the use of GBAs for these type of applications has been performed, including:

- A study into the use of GA's to adapt rules for a battle management system for the AEGIS Combat System [1].
- A similar study [2] using a symbolic rule-based system called SAMUEL.
- A method for selecting an investment portfolio, using GA's for Index Tracking [3].
- An optimal system for planning the Water Distribution Networks [4].

3.2 Scheduling

Many systems involve some form of scheduling. One such example is the Convoy Movement Problem which is detailed later in this paper.

The most common problems encountered are:

- Job Shop Scheduling, (see for example, [5][6]);
- timetabling [7][8];
- the Travelling Salesman Problem (TSP) [9][10].

Job Shop Scheduling involves making the best use of the resources available in some manufacturing process. The problem is usually of the form where several types of product are being manufactured, each of which requires a varying amount of time on each of the available types of machine (e.g. drills, lathes, welding equipment, etc.). The object is to allocate the different jobs to the machines such that various constraints are met, for example:

- items must usually visit machines in a particular order;
- there is a cost involved in changing a machines setup when it swaps to a different job;
- machines should be utilised as fully as possible;
- each item being produced normally has a deadline or quota which must be met.

The TSP is an extremely simple application to define, but is much more difficult to solve. There are a number of *cities* connected by *roads* of different lengths. The problem is to find the shortest possible route which visits all of the cities.

GA and GP literature contains many examples of applying these techniques to scheduling problems, and have proved their suitability in a large number of cases.

These types of problems can crop up in Mission Systems applications directly (as straightforward scheduling or routing problems) or in more complex forms such as the problem of loading convoys for amphibious operations.

This problem has a variety of constraints which must be satisfied simultaneously, for example:

- equipment must be loaded in the reverse order that it will be required during the operation (placing landing craft at the back of a hold may not be a good idea);
- the weight of the equipment should be balanced as evenly as possible (this is even more critical in the case of loading large transport aircraft rather than naval vessels);
- the volume taken up by the equipment, and the dimensions of larger pieces needs to be considered;
- certain items may have special safety requirements (e.g. explosive materials);
- access may be required to certain equipment (e.g. for maintenance tasks while en-route).

These more difficult problems may be more suited to the use of heuristic techniques such as Genetics-based algorithms, simply due to their complexity.

3.3 Autonomy

Again, the use of Genetics-based algorithms for developing autonomous agents or systems has been demonstrated many times in the GA and GP literature.

Various aspects of the problem have been addressed including:

Machine Learning & Classifier Systems - Simulating artificial creatures behaviour is a popular area [11][12].

GBAs have always been highly connected to the artificial ecology community. One of the earliest examples is Wilson's ANIMAT system. This was a simple creature that wandered around a simple landscape, filled with food, trees and empty space (see figure 1). The behaviour of the creature was controlled by a classifier system that mapped the inputs, what the creature could see, to outputs, where the creature moved.

[2] The Defence Evaluation and Research Agency (DERA) is an Agency of the UK Ministry of Defence.

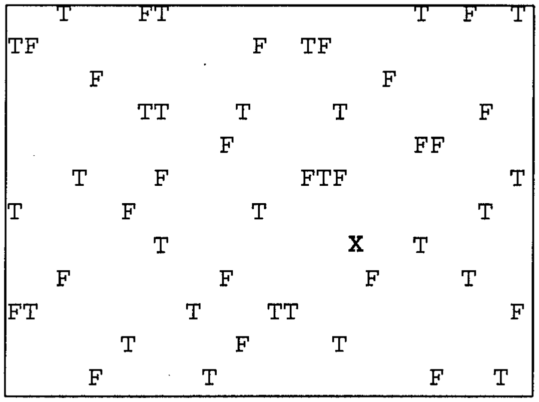


Figure 1 - The ANIMAT world

The inputs were simply determined by the contents of the adjacent squares. These were mapped from a two dimensional display onto a binary input string (as shown in figure 2 below). The binary information represented the input from two simple sensors:

a) sight - returning a 1 for a tree or food, 0 for an empty square;

b) smell - returning a 1 for a square containing food, 0 otherwise.

The behaviour of the Animat (as the creatures were called) was then determined by consulting a classifier system with genetically determined rules. This eventually determined where the animal would move, and what food it would receive. The ANIMAT process contained a number of novel features, including the ability to generate new rules in novel situations, and amalgamate rules with the same outcomes into new generalised rules.

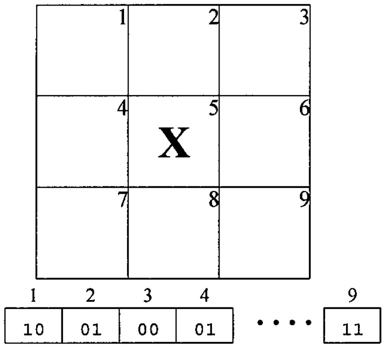


Figure 2 - ANIMAT local sense encoding

The Animat work was very simple, and other autonomous learning experiments have followed. Koza [12] has produced food trail following ants, and there are more complex artificial animals and environments, that have included predators etc.

Game Theory - GAs have been applied to learning successful strategies in the Iterated Prisoners Dilemma (see [13]). Card games have also been taught, and GAs are now used to provide opponents in various computer games.

The Iterated Prisoners Dilemma is one of the core examples of game theory, since it is extremely simple, and yet can lead to complex strategies developing.

The basis of the original game is that there are two prisoners being investigated for a bank robbery. They are interrogated separately. If both *co-operate* with each other, and deny everything, they will both be released. If one prisoner denies the crime, and the other blames the first one, (i.e. the second prisoner *defects* rather than co-operates) then the first prisoner gets a long sentence and the defector gets released *and* given a reward. If both prisoners blame each other, (i.e. both defect) they both go to jail, but for a shorter time. Both prisoners know the rules. This is usually shown in the form of a payoff table, as shown in figure 3.

		Player 2	
	Decision	Co-operate	Defect
Player 1	Co-operate	Both score 6	Co-op scores 0, Defect scores10
	Defect	Co-op scores 0, Defect scores10	Both score 2

Figure 3- Prisoner's Dilemma Payoff Matrix

With the problem as stated, there is only one sensible outcome, if a prisoner co-operates he will score 6 or 0 depending on the other prisoners action, if he defects he will score 2 or 10. Therefore on average he scores better if he defects, and so both should blame the other.

The problem becomes more interesting if it is run repeatedly, and becomes an *iterated* dilemma. In this variant each prisoner remembers what their counterpart did to them in the previous round, and can modify their behaviour accordingly.

In this situation, the best long term strategy for both players, is to co-operate, but there is always the temptation, to defect to grab a short-term gain. This leaves the field wider open for a range of strategies to develop.

It is these strategies that were evolved using GAs in a study by Axelrod. This used competitive selection (as described in section 4.2), .where fitness was determined by playing every strategy against every strategy, including itself. The total payoff was used to rank the fitness selection, and the normal GBA crossover and mutation operators were used to produce new strategies from the old.

The results produced some highly complex strategies, that would use defection to attempt to identify the behaviour of their opponents, and then exploit their weak points. This is one of the most successful examples of competitive selection.

Variants of the iterated Prisoners Dilemma have been applied to such diverse systems as international politics[3], and the ecology of tree heights in forests[4].

Control Systems - Koza has applied GPs to many control

[3] ~~For example, if countries abide~~ by fishing quotas, fish stocks can recover, and in the long term everyone benefits. The temptation to over-fish is ever present however, in order to maximise short-term profits.

[4] Similarly, if all trees grow to the same height they all get an equal share of the sunlight, and benefit equally. Taller trees get a short-term advantage, but in the long-term, if all trees try to grow as tall as possible, everyone loses due to the resources which much be expended to achieve the growth.

systems. Two of the most famous are reversing a truck/trailer combination [14] and Balancing a broom on a cart [15]. Goldberg's original work in the area was a control system, for gas pipelines, which led to the publication of his highly influential book, [16].

The main advantage of GBAs are that they are able to produce robust, adaptive systems, without having to resort to using *brittle* domain specific knowledge.

Systems based on Genetics-based algorithms can continue to function even if the environment they find themselves in changes. Counter-intuitively, it has been found, in a number of cases, that by varying a problem (either through adding noise or having a time variant factor) a Genetics-based algorithm's performance can actually be improved (reasons for this are discussed in sections 4.2 and 4.3).

4. POTENTIAL BENEFITS

4.1 Overview

Genetics Based Algorithms are not a panacea. They can carry a high computational overhead, and it is often better to use a simpler and more efficient technique, such as hill-climbing, or exact methods, if the problem can be solved using analytical methods. Where GBAs are most effective however is in the domains where traditional search techniques are weakest. Problems in these domains are naturally suited to GBA solutions, and they are problems that occur frequently when dealing with the vagaries of the real world.

4.2 Time Varying Problems

Many search techniques depend on the use of a static problem environment. This is why traditional AI methods have so much difficulty producing effective robot controllers. By the time the controller has built up a model of the world to work from, and made a decision based on it, the world has usually changed.

GBAs, by contrast, perform better in the changing environment. Due to the large population involved, they maintain a large and diverse range of potential solutions. If the problem or environment changes, they are much quicker to adapt as they already have a wide range of solutions to apply to a problem.

Time varying the environment can actually increase the performance of the GBA by keeping the evolutionary pressure high.

Once a GBA has reached a stable optimum, and the population has converged on this area to a large degree, the *difference* between the best and worst individuals in the population will be relatively small. There is therefore no selection pressure, and the population begins to stop evolving. A factor known as *genetic drift* sets in, where population members tend to randomly move around the optimal solution. Fitness can actually decrease due to this lack of evolutionary pressure.

A time varying problem can provide the necessary evolutionary pressure to keep the individuals striving to improve.

A useful technique to use in GBAs, is to deliberately vary the environment over time in order to keep the selection pressure (and therefore the rate of evolution) high.

This can be achieved by:
a) injecting noise into the environment;
b) normalisation - i.e. taking the fitness values of the population and multiplying them by a factor which keeps the difference between the best and worst individual at a reasonably constant value.
c) Co-evolution.

Co-evolution

This involves co-evolving the evaluation function, as well as the solutions.

In this case the evaluation function itself is given a fitness value according to how *poorly* it can make the solutions perform, and gradually evolves to be a more and more difficult test.

This has the effect of forcing the solutions to evolve as rapidly as possible just to avoid their fitness deteriorating in the fact of a harsher environment.

This approach has been used in Iterated Prisoners Dilemma problem (see previous section). The possible use of this technique within Mission Systems is outlined in sections 5.6 and 5.7.

Dominance

The simple genetic algorithm uses a haploid representation, a single chromosome. Dominance methods use a Diploid (two chromosome) technique more similar to nature. This approach means that there are two possible values for each location, the gene from chromosome 1, and the gene from chromosome 2. The combination of these values determines which is actually used. The simplest scheme is that if both are identical, then the value is used, but if they are different one is said to dominate the other. This gene is termed dominant, and the other gene recessive.

This method of dominance and recession has been shown (see Goldberg and Smith, [17]) to give great improvements in oscillating non-stationary problems. The theory for why this should be is that the recessive gene allows an alternative problem solution to be stored. The alternative, encoded in the recessive genes, survives in the population, even though it makes no real contribution in the current environment. When the problem changes the recessive genes can quickly re-establish themselves, and the genome will rapidly adapt. Modifications to the scheme include altering which gene is defined as 'dominant'.

There are significant improvements using this method compared to a haploid GBA in these situations, where knowledge of a past solution to a problem is useful in the changing environment.

4.3 Noisy Environments

Real world problems do not tend to have perfect or complete information available. Again, this has often proved troublesome for traditional AI methods with truth-maintenance, as a new piece of information may seem to directly contradict all others. Like Neural Networks, GBAs are resistant to noisy inputs. Noise resistance is a survival characteristic itself, which will be rewarded in successive generations. Therefore there is some selection pressure to develop this secondary characteristic in parallel with the primary ones.

By definition, the best genome in the population will be the member that has solved the task despite the noise. This can be used to advantage, as in [18] where a GA was used with a deliberately high mutation rate (which effectively injects noise into the search).

The population then tended to converge on *reasonably* good areas of the solution space, which were tolerant to single point mutations (i.e. all of the neighbouring locations were also reasonably good), in preference to any narrow optima, where any single mutation would give a poor result.

This approach was used to evolve a robot controller which was robust to any single failure.

Not only are GBA's more robust in the presence of noise, they can actually be improved by it. This is thought to be due to two factors, the changing environment keeps the pressure on the population to evolve, and the noise effectively adds genetic diversity into the population (i.e. similar individuals will produce less similar results).

4.4 Simple Implementation

GBAs are a very simple concept to understand. The basics are easily set up, and a standard GBA can be implemented and applied to any problem which has an evaluation function available, in a matter of hours.

One of the major advantages of this simplicity is the problem independence of the GBA mechanism. GAs are a prime example of this, operating on binary strings in the genome, with no knowledge of the purpose of the string.

This means that the same mechanism can be re-used multiple times, only the evaluation function and initialisation need to be changed. If an evaluation function is available for a problem, which it may well be as it is an essential part of most AI solutions, bolting the mechanism and the evaluation together can produce a rapid prototyping environment.

Unfortunately in some cases, particularly where the problem exhibits a high degree of structure, the fact that GBAs are ignoring this *free* information can lead to performance degradation. In these cases it can be necessary to add in some problem specific knowledge (usually in terms of adapted genetic operators) in order to achieve the required level of performance.

Standard GBA's can be used as a quick method for 'sketching' the problem space (i.e. identifying promising areas). This can then be followed up by applying some other technique to search these areas in more detail, or experimenting with the GBA, in order to improve its performance.

However GBA's never require as much problem specific information as other techniques such as constraint satisfaction, which requires a great deal of analysis and expertise in order to use.

4.5 Complex problems

GBA's tend to be better approaches to problems which are complex. The way in which they search a solution space is an efficient trade off between *exploration* (searching for new optima) and *exploitation* (utilising the knowledge gained so far about the shape of the space to provide reasonable solutions at each stage of the search).

Many other techniques can only be used (or used effectively) if the problem domain can be sufficiently analysed, and understood. This is one advantage of the fact that GBA's do not require specific problem knowledge.

Another advantage is that GBA's do not tend to suffer from an explosion of computational time required, as problems are scaled up, in the same way that many other techniques do.

4.6 Multiple Solutions

GBA's by definition work on populations of solutions. As a search progresses, not only does the best solution found improve, but the average fitness of the population also increases. This means that at the end of a search, the Genetics-based algorithms will automatically be able to provide alternative solutions which are of high quality.

Multiple solutions can also be *explicitly* searched for within the same run, by using methods such as speciation and niche filling. It is possible for a GBA population to converge on two or more high peaks within the same run, essentially producing two answers

This method involves either explicitly dividing the population in some way, or reducing the fitness of a solution by some factor related to the number of other individuals which are close to that point.

This encourages solutions to be *different* and also has the effect of introducing a time-varying effect, which has its own potential advantages as detailed above.

5. EXAMPLE MISSION SYSTEMS APPLICATIONS

5.1 Introduction

This section details some of the Missions Systems work currently being carried out within DERA, where Genetics-based Algorithms are actively being studied.

5.2 Frequency allocation (terrestrial)

Any wide-area radio network whether military (e.g. combat radio nets) or commercial (e.g. cell phone networks), has to deal with the problem of frequency allocation.

There are two main types of problems which may arise from this. Given a network of radio transmitters/receivers, which are varying distances apart (as in fig 4):

a) given a set of distinct frequencies to use, allocate them to the nodes in the radio network such that no links interfere (or, if this is not possible, minimise the level of interference);
b) discover the minimum number of distinct frequencies required to make a zero-interference allocation possible.

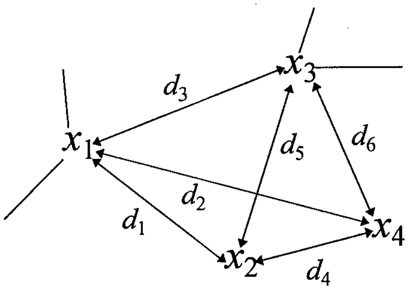


Figure 4 - View of links present in radio network

This was the problem domain which was chosen for the EUCLID CALMA (Combinatorial Algorithms for Military Applications) Project - RTP 6.4.

This was an 18-month study conducted by French, British and Dutch industry/academia, which aimed to investigate the potential of various algorithms to the solution of the more complex combinatorial problems arising in military applications [19][5].

The techniques which were studied included:

a) Genetic Algorithms;
b) Interior Point Methods;
c) Constraint Satisfaction;
d) Simulated Annealing
e) Local Search;
f) Tabu Search.

In order to successfully test these methods a large amount of realistic radio link frequency assignment problems (RLFAP's) were required.

A data set of sample problems was provided by CELAR (Centre d'Electronique d'Armement, France). However this set proved to be too small to meet the needs of the study. It was decided that a method for creating randomly generated examples was needed. However, this can produce problems in itself, such as:

a) can the examples be guaranteed to be realistic;
b) how can the algorithms be accurately benchmarked against a problem for which the optimal solution is unknown (or even if a solution exists).

These problems were solved by developing a software tool, GRAPH (Generating Radiolink Frequency Assignment Problems Heuristically), which generated randomised examples *based on* those contained in the CELAR data set and at the same time recorded a solution to the new problem by extrapolating the appropriate CELAR solution [20].

The RLFAP as used in the CALMA study can be stated as follows:

a) there is a set of links that require frequencies to be assigned to them (typically 200 to 1000 links);
b) there is a set of discrete frequencies available (typically 50);
c) when a frequency is assigned to a link it must satisfy a (large) set of constraints of the form:

   $|f_a - f_b| > s_{ab}$ or $|f_a - f_b| = s_{ab}$

   where f_a is the frequency assigned to link a
   f_b is the frequency assigned to link b
   s_{ab} is separation required between links a and b
d) the objective is to use as few of the available frequencies as possible - or in the cases where there is no feasible solution, to break the minimum number of constraints.
e) the problem may be complicated by assigning weightings to the constraints, to indicate priority.

A quick calculation shows the enormous size of the problem, the *simplest* example, with 200 links and 50 frequencies, has 50^{200} possible solutions.

Two approaches to applying a GA to this problem were attempted, a standard GA (with various modifications trialed), carried out by the University of East Anglia and a variant named a Pyramid Algorithm developed by the University of Limburg.

The standard GA approach performed rather poorly, the conclusion was that it made the task much more difficult for itself by performing a blind search when so much extra information was readily available in the problem structure. Several types of crossover and mutation etc. were studied, in an attempt to improve the search, but overall the algorithm did not perform especially well.

It was recommended that the use of problem specific genetic operators should be investigated and developed, in order to improve on this.

By contrast the Pyramid Algorithm has been specifically designed for this class of problem and performs significantly better.

This algorithm differs from the standard GA in the following way:

a) it starts with a population of 1-optimal solutions (a 1-optimal solution is one which cannot be improved by changing just one of the frequency allocations);
b) consecutive pairs of solutions are taken as parents, and the best child possible is made from each pair (where each frequency allocation is inherited from one of the parents);
c) this child is then 1-optimised (i.e. each allocation is tested in turn to see if changing any one value will improve on the solution);
d) the child then replaces both parents in the next generation (thus halving the population each time);
e) this continues until only one solution remains.

This is a radical departure from the GA, and is more like an Evolutionary Strategy technique.

Constraint satisfaction and the *branch and cut* (based on linear programming) techniques were shown to be the best techniques where there was a feasible solution (and was able to find the *optimal* solution in all cases).

[5] This reference is to a document which reports on the final outcome of the study, and summarises reports generated during the project. These reports and other supporting documentation are available at: http://www.win.tue.nl/win/math/bs/comb_opt/hurkens/calma.html

The drawback with both of these techniques however were that:

a) they were unsuitable for the infeasible cases;
b) they required expert operations research knowledge to develop the problem specific algorithms;

By contrast the Pyramid Algorithm, while unsuitable for the feasible cases, consistently found the best solutions in the infeasible cases (albeit using a large amount of computation effort, when compared to the other approaches).

5.3 Frequency planning (satellite communications)

DERA Defford are currently studying the suitability of various techniques for frequency planning in satellite communications.

This involves assigning frequencies to communications transmitted over the UK's military satellites (satellite accesses). The frequencies used must obey a variety of constraints, such as:

- In general, satellite accesses must not overlap. The exception are certain types of signal modulation, such as CDMA (Code Division Multiple Access), which allow bandwidth sharing to some degree.
- Certain types of equipment may be limited as to the frequencies they can transmit or receive.
- The satellite transponders have a finite amount of bandwidth which can be allocated (typically around 50 to 150 MHz).
- A ground terminal must abide by any local restrictions regarding frequency use (to avoid interfering with local air-traffic control for example).

The main differences between this problem and the terrestrial frequency allocation problem are:

- The large number of possible frequencies (anywhere in the bandwidth to a resolution as fine as 50Hz), typically 1-3 million;
- There is (in general) no frequency re-use involved;
- The need to take intermodulation products into account.

Intermodulation products (IMP's) are interfering signals which are formed when two or more radio signals pass through the same amplifier, particularly when it is operating at 'close' to its maximum output power[6]. The UK military satellites typically carry tens or hundreds of accesses, and usually operate at power levels sufficient to generate IMP's.

3rd Order IMP's (which are the largest and therefore the most problematic) appear at frequencies given by the following equations:

Type 1 $I_1 = 2F_1 - F_2$ $(F_1 \neq F_2)$

Type 2 $I_2 = F_1 + F_2 - F_3$ $(F_1 \neq F_2 \neq F_3)$

where F_1, F_2 and F_3 are the frequencies of any of the desired accesses.

The Type 2 IMP's are typically 6dB larger (i.e. four times as large) than the Type 1, which makes them the more significant of the two.

Thus for two accesses (at frequencies A and B), two Type 1 IMP's are formed at 2A-B and 2B-A.

For three accesses (at A,B,C), 3rd Order IMP's form at:

a) 2A-B, 2A-C, 2B-C, 2B-A, 2C-A and 2C-B (Type 1);
b) A+B-C, A+C-B, and B+C-A (Type 2);

as shown in figures 5 and 6.

In general, there will be: $\frac{n}{2}\left(n^2 - n\right)$ 3rd Order IMP's formed when n accesses share a transponder. This obviously rises rapidly as the number of accesses increases[7].

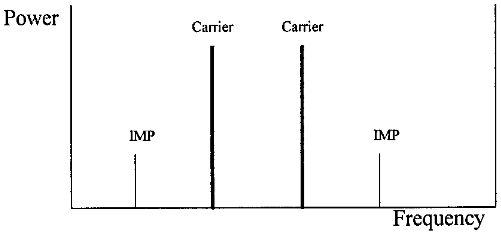


Figure 5 - 3rd Order IMPs with 2 signals

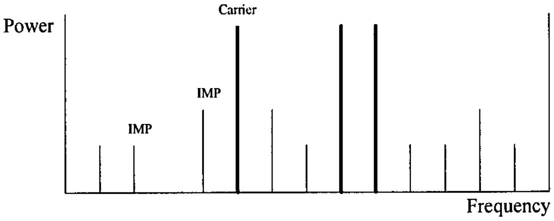


Figure 6 - 3rd Order IMPs with 3 signals

Calculating the power of each IMP is more complex, and needs to take account of: the characteristics of the transponder, how close it is to its maximum output power, and the power levels and types of the signals which contributed to its formation. In general however, the larger the signals involved, the larger the IMP's produced.

This adds two constraints to the problem:

- Large signals must be placed so that their IMP's have minimal effect (i.e. so that they either fall outside of the valid bandwidth of the transponder - in which case they will be filtered out, or so that they arise in unused areas of the available bandwidth).
- Small signals must be placed to avoid overlapping with large IMP's, or their signal to noise ratio will reach unacceptably poor levels.

These constraints are particularly difficult, as adding or moving a single access will often create or shift hundreds to thousands of IMP's.

[6] RF systems are measured on an exponential scale, so 'close' to maximum power may in linear terms mean at power levels greater than 6% or 7% of the maximum output wattage.

[7] This problem becomes more complex if 5th Order IMP's are also taken into account, as while these tend to be slightly smaller, they appear in significantly greater numbers.

Since this is an optimisation problem (i.e. the allocation of frequencies which minimise the levels of interference), GBAs should be suitable for this task. The main difficulty arises in deciding how to represent the problem in such as way as to facilitate an *efficient* search. This is made less easy by the presence of *hard constraints* within the problem.

The study is considering a number of techniques, amongst which are Genetic Algorithms, attempting to directly evolve a frequency plan, and Genetic Programming, to attempt to evolve a function capable of returning a list of values to be used as frequency allocations. The second of these two techniques has been implemented and shows a great deal of promise.

One possibility which may stem from the GP approach is that, by passing in parameters describing the accesses, it may be possible to develop a more efficient search which takes advantage of the extra domain knowledge.

Previous work on minimising the amount of interference produced by IMP's has tended to concentrate on either analytic techniques or algorithms which take a frequency plan and shuffle accesses around in some way in an attempt to improve the interference levels.

This work also tends to concentrate on a simple version of the problem, where all accesses are of the same type (basically, carrier waves), and the bandwidth is divided into equally sized *slots*, which can either hold one access or be empty.

Analytic techniques can provide solutions where the accesses are all free of interference. The price of this is that they require impractically large amounts of bandwidth, in order to allow the signals to be spaced far enough apart.

In a limited bandwidth problem, this technique is not useful except for trivial cases with few accesses.

Work carried out by Okinaka et al [21] developed a practical algorithm for iteratively shuffling accesses around a limited bandwidth (divided into slots) to minimise the interference from IMP's.

The results from the Okinaka study will be used as the benchmark to compare against the new techniques.

The study will begin by:

a) comparison of a variety of GA's, GP's and constraint satisfaction approaches with the Okinaka results (by repeating the same trials for the simplified problem);
b) once the techniques are producing reasonably good results, the problem will be made more realistic by:
   - removing the concept of slots, allowing the accesses to be placed anywhere in the band;
   - taking into account the bandwidth and spectral shape of the accesses;

Once this has been completed and it has been shown whether or not the techniques can be used to produce good frequency plans in the presence of IMP's, the other constraints can be added into the problem. These constraints are, on the whole, much easier to satisfy than the IMP requirements.

Three variations on this problem will be examined:

- the baseline case, where a set of accesses must be placed onto an empty transponder;
- placing a set of new accesses onto a transponder which is already carrying traffic (which cannot be disturbed);
- placing a new set of accesses onto a transponder already carrying traffic, and allowing existing low priority accesses to be moved or deleted if necessary (keeping disruption to a minimum).

All three techniques will be used to calculate answers to a set of example problems and will be compared in terms of speed and the quality of the plans produced. The comparison will need to take into account the fact that constraint satisfaction will give the same answers each time it is run on a particular problem, whereas GA's and GP are not so consistent and the quality of the plans could vary widely on different runs, therefore some sort of statistical analysis will be required.

An initial attempt at the problem using a very simple GP approach has produced some promising results. The GP was set up to simply produce functions made up from the operators + * / -, the integers 1 to 9, and a single variable X.

These functions, f(X) were then evaluated for X=1, X=2 ... X=N, where N was the number of desired accesses. The value returned by f(1) was taken as the slot to place the first access in, f(2) the slot for the second, etc.

The results of various trials showed that even this simple algorithm could quickly evolve mapping functions which created frequency plans which were of a similar standard as those produced by the Okinaka algorithm.

5.4 Convoy movement

The Convoy Movement Problem (CMP)[8] can be seen as a variation of the Travelling Salesman Problem (TSP) - which is probably the most common application to be found in AI, or search and optimisation literature.

The CMP differs from the TSP, in that there are several convoys (as opposed to one salesman), which must move from a number of starting points, to specified target destinations (rather than visit all destinations).

The primary objective is to find a set of paths which achieves this, minimising the overall time taken (i.e. the time between the first convoy starting to move and the last convoy reaching its destination) and meets the following constraints:

- only one convoy allowed at each location at any one time;
- no overtaking allowed;
- convoys may have to reach their destination within a specified time.

A secondary objective is to minimise the total amount of travelling required (i.e. the sum of the travelling times for each separate convoy).

In the initial version of the CMP, no stopping was allowed, once a convoy had set off it would keep moving until it

[8] The Convoy Movement Problem, and the following two studies (route finding over unstructured terrain, and weapon target assignment) are all part of the same overall study, into Technology for Decision Support.

reached its destination. This caused problems where there were bottlenecks in the network of routes available (e.g. bridges), or in the case where many convoys started from the same point (since the above constraints could only be achieved if the number of convoys was less than the number of routes leaving that location).

The CMP had to be extended to allow convoys to wait at a location for some period of time. Some of the locations where defined as blocking (i.e. if a convoy is stopped at this point, no other convoy can pass it) and others as non-blocking (where other convoys can pass through the one which is waiting).

This obviously increases the flexibility in the proposed routes and allows better solutions to be found. Unfortunately it also increases the number of potential solutions and makes the problem significantly more difficult.

The CMP has been tackled by using a GA toolkit, the X-GAmeter from the University of East Anglia. This has allowed various GA's to be tested, and compared for the CMP.

The toolkit allows a GA to be built by picking from a selection of modules for each aspect of the algorithm. For example, there are many feasible techniques for:

- creating individuals in a GA population,;
- selecting individuals for breeding;
- performing crossover;
- replacing individuals within the population.

The toolkit contains functions which allow the user to choose any of the more commonly used techniques for selection etc, and mix and match these modules to easily build a particular type of GA for examination.

The kit also allows the setting of the important GA parameters such as population size, number of trials to perform, crossover and mutation rates, etc.

Work is ongoing on the CMP, comparing the performance of pure GA's with Simulated Annealing, various enumeration techniques and several hybrid GA's (i.e. GA's combined with other techniques).

Initial results would seem to indicate that a hybrid GA approach may provide the best performance for this type of problem.

Future work may include the extension of the problem to include the consideration of:

- forcing convoys to halt every few hours for brief rest periods, and for six hours after every 8-10 hours of travel;
- only allowing convoys to halt at a subset of the locations available (i.e. designated safe areas);
- supplies (designating certain locations as supply depots, and/or having mobile supply convoys - convoys must then visit a depot or supply convoy once every 24-hours);
- re-planning (allowing new information be taken into account, and changes made to the plan as, for example, convoys suffer breakdowns, or discover bridges to be impassable, or routes congested with other traffic, etc).

Within this study, a lot of work has also been carried out into how to best display the solutions to the vehicle commander, and how to allow him to interact with the system for re-planning purposes.

Significant effort has also been put into the problem of how to ensure that the comparison of the various GA's trialed provides accurate and effective information, and how to analyse the results in a rigorous manner.

The process of planning an experiment so that appropriate data (that can be analysed by statistical methods) is collected is called the Statistical Design of Experiments (SDE) [22]. This is an effective method of eliminating known sources of bias and guarding against unknown sources of bias.

This method has been implemented for the experiments carried out in the CMP study, and allows conclusions to be drawn (with confidence) as to which parameters, and algorithm variations (and combinations of these) perform optimally on this problem.

5.5 Route finding over unstructured terrain

Route finding in a battlefield scenario, over unstructured terrain is an extremely complex problem. Many factors must be taken into consideration, including

- safety of the route (including potential visibility to enemy observers);
- time taken to negotiate the route (and the vehicles capability to pass over the specified type(s) of terrain);
- fuel consumption.

There are algorithms, already in existence, which can perform this task in real time, and produce optimal or near-optimal routes. These algorithms suffer from the fact that the amount of computational effort required is proportional to the number of data points being considered (i.e. doubling the area covered by the map, or increasing the resolution, will double the amount of time required).

One of these algorithms was extensively used in field trials as part of the VERDI-2 project, using a dedicated parallel computer and limited to a restricted operating area (to ensure that the response times were adequate) [23].

The conclusions from these trials were that the tool was extremely useful, but it only produced one route, and it would be much more useful if it could suggest a set of feasible routes. This would allow multiple vehicles to make the same journey via diverse, alternate routes, and allow the vehicle commander to use any local knowledge, not available to the system, in order to select one route over another.

Since Genetic-based algorithms maintain a population of solutions, they have the potential to fulfil this requirement without significantly increasing the time taken to generate the solutions.

An initial GA has been implemented to attempt this problem, and has shown that this approach has definite potential in this problem area.

The GA represents a route by a set of waypoints (with the co-ordinate of each point converted into binary form), and gradually evolves these waypoints into better routes.

The current GA starts with a set of n waypoints (where n is relatively small) and begins to evolve routes. At specific time intervals (i.e. after set numbers of generations) the number of way points is doubled, effectively allowing the route to be defined to a higher level of resolution.

This is achieved by inserting a new way point in-between each of the existing ones. Therefore for a route defined by waypoints: ABCDEF, a new route would be created AxBxCxDxExF where the x's indicate new sets of co-ordinates.

The new waypoints can be selected in a variety of ways, such as simply randomly inserting co-ordinates (possibly constraining the choice to general areas between the bracketing waypoints) or selecting a point on a straight line between the two existing waypoints, etc.

This is done to allow the solutions to be refined as the search progresses. Putting too many waypoints into the algorithm at the start of the search, before it has had chance to make a rough cut of the more likely areas for routes, slows down the performance of the algorithm significantly.

The fitness function to assess the routes against has proved to be more complex than it may appear at first sight. It is obvious that the function must assign a cost (made up from assessing all of the factors such as fuel, time, safety, etc and applying appropriate weights) to each route, but how should it regard the route *in between* the waypoints?

The simplest option is to assume the vehicle naively takes the shortest straight line route from one waypoint to the next. A more satisfactory option would be to calculate the lowest cost route possible between the two waypoints - but this makes the problem recursive, and very time-consuming.

Several functions have been tested which are compromises between the two extremes, but this is one area which will need to be investigated further.

Future work will also continue to develop the GA, and investigate the possibility of implementing a GP technique.

5.6 Weapon Target Assignment

Weapon target assignment is an optimisation problem which attempts to use the defensive resources available to a commander, in such a way as to maximise their utility against some threat.

There are actually two problems here:

a) Tactical weapon target assignment
  - Here there are a number of known incoming threats, defined by a trajectory, and a value related to the strength of the threat it poses. The defender has a set of weapons with which to attempt to destroy the threat before it reaches its target. These weapons can be given characteristics such as number of shots available, speed, accuracy, minimum time between subsequent shots, etc.

b) Strategic weapon target assignment
  - This involves the deployment of weapon systems and sensors, to defend a set of assets against a set of potential types of attack.

Heuristic techniques such as GBAs, due to their stochastic nature, would appear to be unsuitable for the tactical problem because of the short response time which is critical for this type of system.

Various other types of approach have been studied previously and shown to perform well, including branch-and-bound, greedy hill-climbing, linear programming and neural networks.

However, all of the studies to date have concentrated on the problem where the defensive commander has full knowledge of the threat posed to the assets for which he is responsible. In practice this is unlikely to be the case, and in particular, as the defensive commander plans further ahead in time, his knowledge of the situation will diminish.

The exact methods, which perform so well when provided with full knowledge of the situation, are likely to degrade significantly in performance under these conditions. They will continue to produce 'optimal' solutions, unfortunately these will be optimal solutions to situations which don't actually exist in the real world.

It is possible that the quality of the solution may degrade gracefully (i.e. become less optimal in proportion to the degree of noise/uncertainty in the knowledge it is provided with), in which case the exact methods will still be useful, but this is by no means guaranteed.

In these *real-world* situations heuristic techniques such as Genetics-based Algorithms will have a distinct advantage over the exact methods. Therefore the DRA study into the weapon target assignment problem recommends that these noisy environments should be studied in more detail, and that a combination of some heuristic technique hybridised with an exact method (to improve the response time) may prove to be the most useful approach.

As far as the Strategic problem goes, Genetics-based Algorithms may have even more to offer, since the tight time constraints are not present.

One possible approach to this problem would be to co-evolve both defensive and offensive commanders.

Offensive commanders would be given a defined set of weapons, and have to deploy them against a set of targets of varying value. Defensive commanders would have to deploy sensors and defensive weaponry to attempt to maximise the protection provided to their assets.

Each offensive option could be assessed against each of the defensive commanders in turn, and given a rating as to how well it performed overall, and vice versa for the defensive options.

This would lead to offensive and defensive strategies being evolved which were effective against as wide a range of potential opposing strategies as possible.

The benefit of having opposing populations co-evolve, generating evolutionary pressure to spur on continual improvement has been demonstrated many times in GA and GP research.

An example is [24] where a set of randomisers (i.e. functions which returned 'random' sequences of numbers) were evolved and assessed against a set of functions which tried to predict the next number in the sequences.

Other examples can be found in Game Theory, such as [13], which looked at evolving strategies for the iterated prisoners dilemma problem.

Some similar problems to weapon target assignment have been attempted in GA and GP studies such as the example given by Koza of lizards feeding on insects[25].

In this problem a lizard sits on a branch and observes insects, which appear with a random distribution, in a 180 degree arc in front of him, and disappear again after a varying amount of time (see figure 7). It takes longer to move towards a distant insect in order to eat it, therefore, attempting to catch an insect has a varying degree of success depending on how far away it is.

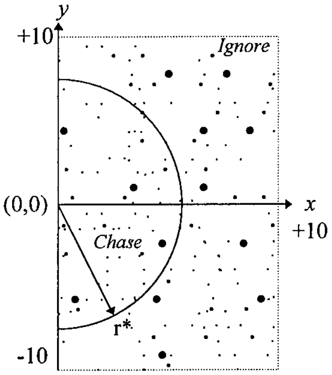


Figure 7-Hungry lizard decision zones

If the lizard waits for insects to appear very close to his perch, he may starve, as this is not likely to occur often. On the other hand, attempting to catch every insect is also a poor strategy, as the distant insects are likely to get away, and waste a large proportion of the lizards time, when other better targets may have presented themselves and been missed.

The object is to find an optimal strategy for the lizard which nets him the most insects.

Koza demonstrated that a GP approach could evolve strategies which closely approached the optimal solution.

5.7 Battlefield simulation

The High Level Operations using Cellular Automata (HiLOCA) project is carried out at DERA Malvern [26]. Its task is to examine the effects of different RISTA (Reconnaissance, Intelligence, Surveillance and Target Acquisition) information on the outcome of battles.

This requirement is easily stated and conceptualised, an example would be the question, 'does adding a radar unit increase the effectiveness of these units'. However, it is difficult to present an abstract way of determining this, and still relate it to the real world. The solution determined at Malvern was to actually simulate the battlefield, and the RISTA information passing back and forth between the various units as they fought. By changing the RISTA configuration, different outcomes would be seen.

HiLOCA uses a units modelled as autonomously acting Cellular Automata (CA), linked to the others by communications links passing RISTA, with a typical command and communications model determining configuration.

Units on the battlefield can:

a) move, subject to terrain constraints and formation;
b) sense, depending on the configuration of the unit, with at least line-of-sight represented;
c) fight, if there is a hostile target in range.

Each level in the command and control hierarchy reviews its own information, and then sends a summary to the next level up, after a short delay. 'Soldier' units pass information to 'Lieutenants' and 'Captains', who may have their own sensor information to add, and are present at the battlefield. This is then passed on to 'Majors' and again up the command hierarchy to 'Colonels'. These higher up echelons spend more time determining orders and building up a battlefield picture. Orders then propagate back down the hierarchy, subject to analysis delays.

Units' behaviour is determined by a mixture of their current situation, from subordinates sitreps and their own sensor information, and orders from higher ranking units. This means that the units do not have to wait for higher level orders before they can act, but will act according to their current orders until changes are made.

Decisions are made on the basis of 'threat maps'. Each available enemy unit is assigned a threat value, based on its size, strength, range etc. These are then divided into eight regions, based on the compass direction, and the total threat in each segment found. The result of this map is to alter the behaviour of the units. Combat is automatically initiated where possible, so only movement orders are considered. These can tell a unit to switch to its own local movement, instead of moving with the group it is attached to, and can indicate direction, either directly towards a threat, an attempt at flanking, maintaining a constant distance (go around) or move away.

The core of the CA are the rules used to convert the threat maps into the orders. In the original system these rules were created by extracting knowledge from military commanders, and were fixed. This has the problem of inflexibility. In effect, the model is constantly playing against a single red opponent, and the blue forces cannot change their *doctrine* in order to make full use of any changed or new RISTA information.

This would in effect bias any results regarding the effectiveness of various RISTA options, in terms of a single combination of red/blue strategy.

One possible way to overcome this is to increase the power of each battlefield agent, and the range of strategies played against, by allowing adaptive rule systems.

This would allow the blue forces to change their long-term strategy and rules of engagement (by adapting its rule sets) to take full advantage of the current RISTA system, and allow the red forces to gradually evolve responses to any differences this might make to the battlefield engagements.

A potential approach to providing this adaptive behaviour is through using a genetically based classifier system.

Classifier systems have been used for a long time in the AI community, they are a constrained version of the typical condition..action rules found in knowledge based systems. They encode rules in a series of `if <some condition> then <some action>`, with the limitation that rules must be a certain fixed length. This makes them perfect for genetic learning

Classifier systems are in many ways similar to cellular automata. There is a list of sensors information, a memory, a store of classifiers (the rules which map condition to action) and a series of available actions.

The system works by taking in information through the sensors, and comparing it with the rules it has stored. One or more rules may match the sensor information, suggesting a number of courses of action. These are then chosen between, and one action implemented. To allow more complex behaviour than this stimulus-response activity, actions can place symbols in an internal memory, which can cycle through the process again.

Learning comes into the system by allowing certain weights to be attached to rules, which determine the rule to use when several are suggested by the situation. The results of the action can then be fed back into the system, good results imply the use of good rules, and poor results mean poor rules.

Genetic Algorithms provide a method for generating the rules. The 'quality' of a rule, determined by how successfully it has been applied, can be used as a fitness function. This allows the normal genetic operators of crossover and mutation to apply between rules, combining aspects of the previous rules to make new ones. If these new rules are successful, then they will prosper, otherwise they will die out and be replaced by more successful rules.

This effect can further be strengthened in this application, as the same technique can be applied to both the red and blue teams. This allows co-evolution to occur in both groups, resulting in an evolutionary 'arms-race', where both teams improve due to the evolution of their opponent.

6. CONCLUSION

Genetics-based algorithms are a powerful search technique, that have been applied in a wide range of domains. Their particular strengths include being able to cope with the complex, deceptive and noisy problems that traditional search techniques have problems with.

They can supply a range of multiple, diverse solutions to a problem and can display high levels of robustness in dealing with unexpected or noisy situations.

The approach needs surprisingly little in the way of brittle problem representation, as the methods applied are generic.

There are a vast number of military application domains in which GBAs can help, including:

a) planning and scheduling resource usage;
b) developing optimal strategies;
c) autonomous agents within simulations;
d) robust, adaptive, autonomous systems (by combining the work from the machine learning, imaging, and control systems);

This paper has barely begun to describe the range of applications that are suitable for Genetics-based Algorithms . Other examples include the use of GBA's in the design of systems such as the construction of Radar antennas[27] or aircraft design [28].

The approach is not perfect. In cases where a problem exhibits a high degree of structure, the fact that Genetics-based Algorithms do not take into account any domain knowledge, can lead to performance degradation, when compared to *problem specific* techniques. In these cases it can be necessary to add in some problem specific knowledge (usually in terms of adapted genetic operators) in order to achieve the required level of performance.

However, many other techniques can only be used (or used effectively) if the problem domain can be sufficiently analysed, and understood; and so this can be an advantage if the problem is so complex as to prohibit this.

Another advantage is that GBA's do not tend to suffer from an explosion of computational time required, as problems are scaled up, in the same way that many other techniques do.

The use of co-evolution can be used to 'spur on' a GBA and improve its performance significantly. This also has the advantage of developing robust, *general* solutions which can deal well with new or changing conditions. This particular approach tends to be particularly appropriate to many military applications (as there is almost always an opponents view to be considered at some stage).

If the studies discussed here are any indication, it would appear, that one way or another Genetics-based Algorithms, will be part of, or affect the development of many future military systems.

7. REFERENCES

1. M.J. Kuchinski. Battle management systems control rule optimisation using artificial intelligence. Technical report: NSWC MP 84-329, Naval Surface Weapons Centre, Dahlgren, VA, 1985

2. Grefenstette, J. Competition based learning for reactive systems, in Proceedings of a DARPA Workshop on Innovative approaches to Planning, Scheduling and Control, San Diego, 1993.

3. Shapcott, J Index Tracking: Genetic Algorithms for Investment Portfolio Selection, Edinburgh Parallel Computing Centre (EPCC), Report number EPCC-SS92-24, 1992.

4. Simpson, AR Optimatics: Optimal design of water distribution systems using GA optimisation - an outline of background and proven experience, from: http://www.adelaide.edu.au/Luminis/Optimatics, 1995.

5. Davis, L. Job shop scheduling with Genetic Algorithms, in Proceedings of an International

Conference on Genetic Algorithms and their Applications, San Mateo, 1985.

6. Hsiao-Lan Fang et al. A promising Genetic Algorithm approach to Job-shop Scheduling, Rescheduling and Open-shop scheduling problems, in Proceedings of the 5th International Conference on Genetic Algorithms and their Applications, S Forrest (ed), San Mateo, 1993.

7. Colorni, A et al. Genetic Algorithms and highly constrained problems: The Time-Table case, in Proceedings of the 1st International Workshop on Parallel problem Solving from Nature, Dortmund, Germany. Lecture Notes in Computer Science, 496. Springer-Verlag.

8. Burke, EK. Automated scheduling of University exams, in Proceedings of IEE colloquium on Resource Scheduling and Large Scale Planning Systems, Digest number 1993/144, 1993.

9. Goldberg, DE and Lingel, R. Alleles, loci and the travelling salesman problem, in Proceedings of an International Conference on Genetic Algorithms and their Applications, San Mateo, 1985.

10. Suh, SY and Gucht, DV Incorporating heuristic information into genetic search, in Proceedings of the 2nd International Conference on Genetic Algorithms and their Applications, Cambridge, MA. Lawrence Erlbaum. NJ, 1987.

11. Wilson, SW. Knowledge Growth in an Artificial animal, in Proceedings of an International Conference on Genetic Algorithms and their Applications, San Mateo, 1985.

12. Koza, JR. Hierarchical automatic function definition in genetic programming, in Proceedings of the Workshop on the foundations of Genetic Algorithms and Classifier Systems, Vale Colorado, Morgan Kaufmann, 1992.

13. Axelrod, R. The evolution of strategies in the iterated prisoners dilemma, in Genetic Algorithms and Simulated Annealing, London, Pitman. 1987.

14. Koza, JR. A genetic approach to finding a controller to backup a tractor trailer truck, in the Proceedings of the American Control Conference, 1992.

15. Koza, JR and Keane MA. Cart centring and broom balancing, by genetically breeding populations of control strategy programs, in Proceedings of the International joint Conference on Neural Networks, Washington, Erlbaum 1990.

16. Goldberg, DE. Genetic Algorithms in search, optimisation and machine learning, Addison-Wesley, 1989.

17. Goldberg, DE and Smith, RE. Non-stationary function optimisation, using Genetic Algorithms with dominance and diploidy, in Proceedings of the 2nd International Conference on Genetic Algorithms and their Applications, Cambridge, MA. Lawrence Erlbaum. NJ, 1987.

18. Thompson A. Evolving fault tolerant systems, in First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA), Sheffield University, 1995.

19. West CL. Combinatorial Algorithms for Military Applications (CALMA) DRA report (Unpublished), 1996

20. van Benthem, HP. GRAPH - Generating Radiolink Frequency Assignment Problems Heuristically, Thesis, Dept of Statistics, Probability and Operations Research, Delft University of Technology, 1995.

21. Okinaka, H. Intermodulation Interference-Minimum Frequency Assignment for Satellite SCPC Systems, IEEE Transactions on communications, Volume com-32, #4. 1984.

22. Fisher, RA The Design of Experiments, Hafner Publishing Co. New York, 1966.

23. Webber, HC VERDI-2 route planning, DRA report (Unpublished).

24. Jannink, J. Cracking and co-evolving Randomisers, in "Advances in Genetic Programming", Kinnear KE (ed), MIT Press, 1994.

25. Koza, JR Genetic Programming 2: Automatic discovery of re-usable programs, MIT Press, 1994

26. Dodd, L and Richardson, SB HiLOCA - Model of High Level Operations using Cellular Automata: URD, DRA report (Unpublished) 1996.

27. Chambers, B et al Application of genetic algorithms to the optimisation of adaptive antenna arrays and radar absorbers in First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA), Sheffield University, 1995.

28. Obayashi, S and Takanashi, S Genetic algorithm for aerodynamic Inverse Optimisation problems, in First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA), Sheffield University, 1995.

8. ACKNOWLEDGEMENTS

The authors would like to thank the following people for contributing information to be included within this paper:

H C Webber and S A Harrison, DERA Malvern

S B Richardson and L Dodd, DERA Malvern

The mission systems applications research mentioned in this paper was funded by MoD(UK). The paper presents the opinion of the authors and may not represent the view of MoD (UK). © British Crown Copyright (1998) Published with the permission of the Controller of Her Britannic Majesty's Stationery Office.

Airport Traffic Management based on Distributed Planning

D. Böhme
Institute of Flight Guidance
DLR – German Aerospace Center
Lilienthalplatz 7, D-38108 Braunschweig

ABSTRACT

Against the background of permanent growth of air traffic the development of planning principles, algorithms, and systems for the Air Traffic Management (ATM) has become an important field of air traffic research. Whilst present ATM planning systems were rather "custom-built" future planning systems will have to be cooperative. This will become true especially for Airport Traffic Management (APTM), since here ground, arrival, and departure traffic flows have to be synchronized.

The paper deals with methods and algorithms that where developed in order to realize a cooperation of distributed planners within the context of APTM.

Although considerations are done on example of the DLR´s[1] TARMAC concept for an ASMGCS it is the intention to treat this subject on a level of abstraction that supports an application on other domains. For the same reason some classification aspects of the interdependence of planning tasks are given.

Two examples of use, distributed, cooperative taxi planning and cooperative planning of runway occupancies for arrivals and departures, are explained in detail. In one example the making of cooperation is considered under the requirement of only minor modifications of an already operating arrival planning unit.

1 Introduction

Against the background of permanent growth of air traffic the development of planning principles, algorithms, and systems for the Air Traffic Management (ATM) has become an important field of air traffic research. Up to now only a few planning systems were successfully implemented in certain areas of ATM. Further systems will be added in future.

Whilst present ATM planning systems were rather "custom-built" future planning systems will have to be cooperative. This will become true especially for Airport Traffic Management (APTM), since here ground, arrival, and departure traffic flows have to be synchronized. The benefit that will be reached for the overall ATM/APTM system will not only depend on the features of the individual planning systems but also on the methods cooperation to be performed, among the individual planning units.

Therefore the design of ATM/APTM planning systems will have to be done under consideration of concepts of Distributed and Cooperative Planning. On the other hand, ATM is a traditional research and application field for planning concepts, since the domain characteristics cause certain requirements which are difficult to meet.

The paper deals with some new methods and algorithms that realize a cooperation of distributed planners. The design problem is also considered under the aspect that presently or in future existing planning systems should be made cooperative without the need of substantial modifications of these systems.

Although considerations are done on example of the DLR´s TARMAC concept for an ASMGCS it is the intention to treat this subject on a level of abstraction that supports an application on other domains. For the same reason some classification aspects of the interdependence of planning tasks are given, too.

The paper is organized as follows: The next chapter shortly describes the basic management tasks and their interrelations of the application area. Requirements and objectives of the system design of a system of cooperative planners are outlined in chapter 3. Chapter 4 describes the cooperation among planning units of APTM. Distributed, cooperative taxi planning and cooperative planning of runway occupancies is treated in detail. In chapter 5 an outlook about further investigations is given.

[1] German Aerospace Center

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

2 Basic Management and Planning Tasks of Airport Traffic Management

Within the context of APTM the generic term management comprises the set of tasks (functions), namely planning[1], guidance, surveillance, and control [1], which have to be performed to ensure a safe and efficient traffic flow. Considering each of these functions and especially the planning tasks (tab. 1) it is important to notice that different parties are involved, namely different ATC units, airport operation managers, and airlines' operation centers.

Although all parties share a common interest in reliable, punctual traffic flow, they perform their planning under individual objectives using specific, but incomplete information about the (whole) system state. On the other hand the various planning tasks cannot be done independently by isolated planning units[2], since implementation of plans within one subsystem might have an effect on states of other subsystems as well, and might lead to constraints which have to be met. For instance, the implementation of taxi plans in a certain area (ground movement area or apron) leads aircraft to certain destinations where they are handed-over to the other area. In this way taxi plans of both planning units have to be concatenated.

Besides these technical reasons there is a fixed ranking of management tasks. This "master slave structure" of the management units is well-founded on safety aspects and often set by operational rules. The two most important cases are:

❒ subordination of runway management to arrival management ("landings before takeoffs")

❒ subordination of ground movement management to runway management ("landings and takeoffs before runway crossings")

In order to carry out a management/planning task within APTM it is therefore necessary

❒ to take into account operational rules of subordination of several management units

❒ to consider not only actual (traffic) situation but also future or planned states of the subsystem environment (see fig. 1)

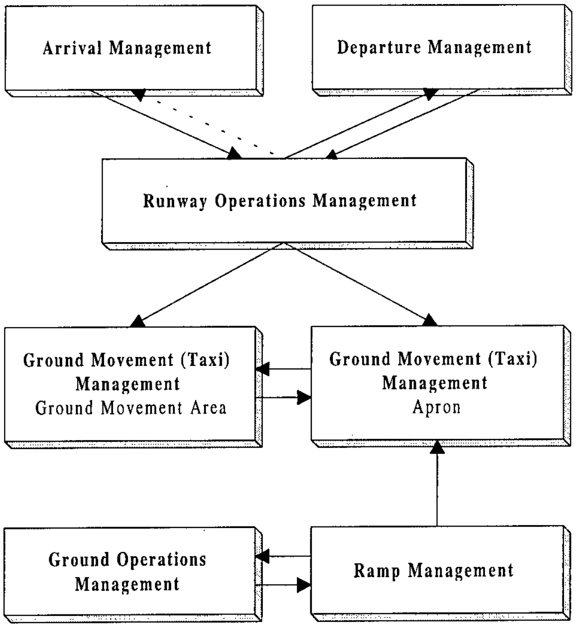

Arrival Management
Departure Management
Runway Operations Management
Ground Movement (Taxi) Management Ground Movement Area
Ground Movement (Taxi) Management Apron
Ground Operations Management
Ramp Management

Fig. 1: *Basic Management Units of Airport Traffic Management and their Consideration of other Units' Plans (e.g. taxi planning on apron has to consider gate allocation, planned by Ramp Management)*

From the viewpoint of automation and of the development of (computer) systems that support planning two consequences result from the explained APTM domain characteristics:

1) Planning can only be done by several, distributed planning units (planning systems), where
   ❒ complexity of the planning problem can be surmounted under time stressed decision demand
   ❒ local authorities for a certain management task retain the leadership and responsibility by having an influence
      ❒ on plans[3] and
      ❒ on constraints as well as on the planning (optimization) function[4]

2) The distributed planning units should be capable to cooperate with other units [2][3].

Whilst early systems were already implemented or are presently under development [4][5] which meet 1), the second requirement has become more

[1] The designations of management functions might slightly differ between different fields. In the ICAO Manual of ASMGCS currently the term routing is use as synonym for planning.

[2] The meaning of the terminus "unit" differs depending on context. In order to avoid an overloading of the term "system" it is used as a synonym when really the technical (computer-) system is meant. Otherwise it designates a certain part of the APTM which performs the planning/management task.

[3] also called direct influence [9]

[4] also called indirect influence [9]

important since the development of "Advanced Surface Movement Guidance and Control Systems" (ASMGCS) is regarded as a need to cope with the growing traffic demand. Against this background several European and national projects were initiated [6][7][8][9][10].

Management Unit	Task Description		Performer
	General	Planning	
Arrival Management	arrangement of arrival traffic flow	scheduling of aircraft on metering fixes and on final approach	ATC-Approach; ATC-Tower
Departure Management	arrangement of departure traffic flow	planning of parameters of departure routes and takeoff procedures	ATC
Runway Operations Management			
❐ Runway Configuration Management	assignment of a specific set of runways that are used for landings and/or takeoffs	planning of an event that triggers the change to the new configuration	ATC-Tower
❐ Runway Allocation Management	assignment of a specific runway for each departure and arrival	no planning task	ATC-Tower
❐ Runway Occupancy Planning	see planning	planning of runway occupancies (time windows) for departures	ATC-Tower
Ground Movement Management	arrangement of ground traffic flow	planning of taxi routes and timed actions (that could be instructed by the controller), a pilot has to do, like e.g. stop, continue taxiing etc.	
❐ Movement Area	... on ground movement area	... on ground movement area	ATC-Tower
❐ Apron	... on apron(s)	... on apron(s); additionally planning of push-back times and schedules	ATC-Tower and/or Airport Control
Ramp Management	gate management	planning of gate occupancies	Airport Control
Ground Operations Management	management of aircraft servicing	planning of the use of different resources (staff, ...)	Airlines

Tab. 1: *Survey of basic management and planning tasks of Airport Traffic Management*[1]

3 Making of Cooperative Planning

The design task comprises the development of adequate cooperation method/algorithms and architectures suitable for their implementation. However, before treating this points more detailed the problematic nature of the establishment of distributed (autonomous), but cooperative planning should be considered, in order to give some clues whether and/or under which circumstances a certain method/algorithm is applicable.

[1] More detailed information can be found in [9] [13].

3.1 Basic Design Tasks

There are two main types of the development task:

a) All or some of the planning units to be made cooperative are already in operation. The development of the existing units was mainly done under consideration of the specific requirements of the respective planning task. In order to disturb the operational process as less as possible and with regard to the investments made software (system) modifications should only be limited and must not affect the system core.

b) The capability to cooperate (to be cooperative) with certain other planning units is already a requirement of initial system design. This will be fact in many cases when several units shall deal with similar planning tasks, but plans will be made for different areas or agents (aircraft) and may result from the organizational structure of the overall management (for instance: ATC for en-route with several, neighboring sectors) or may result from a decomposition of a very complex planning task not resolvable in time.[1]

3.2 Objectives of Cooperation

The objectives of cooperation are closely related to

a) the causes of the interdependence of the planning tasks

b) the degree of the need for cooperation

c) the required quality of cooperation

3.2.1 Causes of Interdependence of Planning Tasks

As already explained in chapter 2 on the example of APTM the two main reasons for interdependence of planning tasks are: the use of common (shared) resources and/or a unidirectional or reciprocal influence on planning conditions.

If they compete for common resources then cooperation has the objective of controlling the access to these resources under the viewpoint of a desired behavior of the overall system. In the other case cooperation should ensure that plans of different units fit together. However, in every case the given organizational structure, i.e. the hierarchy of the units, has to be considered by any cooperation method to be applied.

3.2.2 Degree of the Need of Cooperation

Another aspect for categorization of conditions of the design task is whether

- ❒ the planning units can solve their tasks independently (voluntary cooperation) or
- ❒ if they are forced to work together (compelled cooperation) either
    - ❒ because planning units need other units in order to found a own or common plan, as it was assumed in many examples for illustrating constraint propagation algorithms [11] or
    - ❒ units are principally able to solve their planning tasks by themselves, but cooperation is necessary to adapt plans to constraints that were caused by other planning units

3.2.3 Required Quality of Cooperation

In case that existing planning units are to be made cooperative, an "improved" behavior of the overall system is granted. Against it the much more stronger requirement for an optimum cooperation is of theoretical interest, at least. Both cases may be considered formally in the following way:

Let be $Q_i(P_i)$ the planning function (an optimization function which should be minimized) of unit i and

$$P_i^* = \arg \min_{P_i \in \Pi_i} \{Q_i(P_i)\} \qquad (3\text{–}1)$$

an optimum plan from the set of possible plans Π_i.

Assuming that there exists an overall cost function of overriding importance

$$Q = Q(P_i, \ldots, P_n) = Q(P) \qquad (3\text{–}2)$$

with

$$\begin{aligned} P^{**} &= \arg \min_{\substack{P_1 \in \Pi_1 \\ \vdots \\ P_n \in \Pi_n}} \{Q(P)\} \\ &= (P_1^{**}, \ldots, P_n^{**}) \end{aligned} \qquad (3\text{–}3)$$

then in general

$$P^{**} \neq (P_1^*, \ldots, P_n^*) \qquad (3\text{–}4)$$

[1] An example is treated in section 4.1.

or

$$E\left\{Q\left(P^{*}\right)=Q\left(P^{**}\right)\right\}<1 \qquad (3\text{–}5)$$

holds. In other words: The best overall plan P^{**} usually cannot be achieved by uncooperative planning. On the other hand: Often it is too hard or impossible to find the optimum solution by cooperative planning, but cooperative plans P_c^* should be "better" than P^*, therefore:

$$E\left\{Q\left(P^{*}\right)\right\}\leq E\left\{Q\left(P_C^{*}\right)\right\}<E\left\{Q\left(P^{*}\right)\right\} \qquad (3\text{–}6)$$

3.3 Basic Architectures

The cooperation between planning units can be performed directly[1] [12] or with the help of a supervisor system or function [13]. Accordingly there are two main architectures shown on example of a pair of planning units by figures 2a) and 2b).

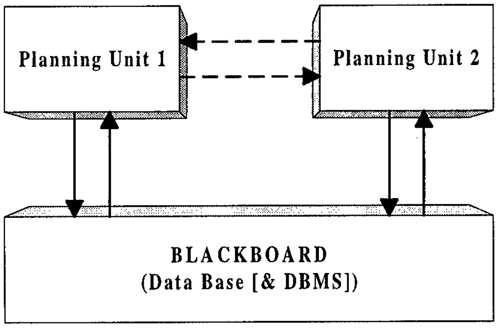


Fig. 2a: *Direct Cooperation of two Planning Units (solid lines: Blackboard Architecture)*

In order to make the cooperation of a set of planning units possible, to enable an asynchronous, autonomous work of the units the use of a so-called blackboard is often preferred and the architectural structure is termed as blackboard architecture, which is well known since the 80ths [10]. The blackboard itself can be realized by any dynamic database, which might be handled by an database management system (DBMS).

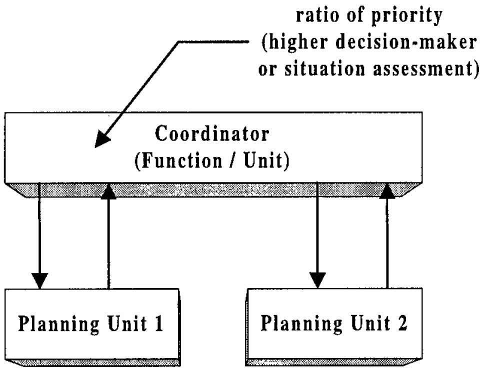


Fig. 2b: *Supervised Cooperation Between two Planning Units*

The alternative, supervised cooperation needs a coordinator function or a unit with a human machine interface (HMI) in case a higher decision-maker should enable to control the ratio of priority ranks between the planning tasks by changing a special tuning parameter (sect. 4.2.2.2). Two things seem to be remarkable:

- ❒ The tuning parameter could also be varied automatically, e.g. by a classifier which chooses a pre-defined value according to a set of situation features or any other situation assessment algorithm.
- ❒ The use of a DBMS as data interface between the coordinator and the planner is advantageous especially during the development period as first experiences made in the TARMAC- and DEFAMM-projects have shown.

The knowledge how to process the data on the blackboard has to be part of the planning algorithms of the units in case of direct cooperation, whereas in case of supervised cooperation the coordinator takes care of information management.

It should be mentioned that in case of cooperation between technical systems the aim of information exchange is mainly to influence rather than to inform the other system (as it might be the case when humans are cooperating). Further a fair behavior should be assumed, so that other units can trust the information.

[1] Here this term is used even when data are exchanged through additional means.

4 Cooperation Among Planning Units of APTM

4.1 Distributed, Cooperative Taxi Planning

4.1.1 Planning

According to the concept of an ASMGCS the traffic flow on the airport surfaces should be improved by application of computer supported (automatic or semi-automatic) ground movement planning. Ground movement planning comprises the planning of all actions aircraft have to carry out in order to reach their destinations on the ground in an optimum way under consideration of actual and predicted traffic situations as well as all constraints which are based on technical and operational reasons.

When assigning all actions to aircraft taxi plans result. Every single taxi plan consists of a taxi route (path) and a set of actions, like *start taxiing, hold, continue taxiing* etc., which should be done at planned times or on certain conditions. Taxi plans have to be made available within certain time windows (planning necessity), i.e. early enough that

- ❒ destinations can be reached in (desired) time and
- ❒ that the controller is able
  - ❒ to notice, to accept or if necessary to correct or refuse plans
  - ❒ and in case there is no data link to give the necessary guidance instructions.

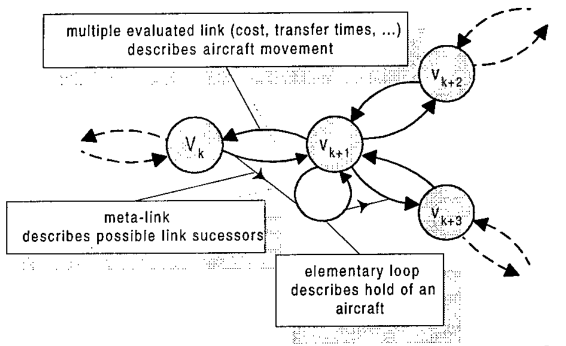

multiple evaluated link (cost, transfer times, ...)
describes aircraft movement
V_k
V_{k+1}
V_{k+2}
V_{k+3}
meta-link
describes possible link sucessors
elementary loop
describes hold of an aircraft

Fig. 3: *Topological Model for Taxi Planning in the Form of an Graph G=(V,E,M,C). V: set of knots; E: set of links; M set of meta-links; C: set of multiple link evaluations*

In every case, taxi planning for a single aircraft has to consider all earlier established plans of other aircraft since all aircraft share the common resource "space". According to the TARMAC concept (DLR's approach to an ASMGCS), automatic taxi planning is based on a suitable topological model of an airport in the form of an graph (fig. 3) with multiple evaluated links which are unavailable for a certain period when occupied by other aircraft.

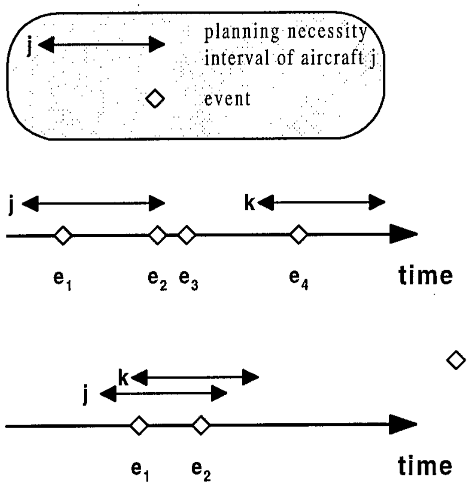


Fig. 4: *Patterns of taxi planning. (above: subsequent planning; below: distributed planning)*
Events of subsequent planning:
e_1: deliverance of taxi plan for aircraft j;
e_2: acceptance of taxi plan by controller;
e_3: change the set of constraints by adding taxi plan of aircraft j ($C \cup P_j$);
e4: planning for aircraft k under consideration of C
Events of distributed planning:
e1: distributed planning for aircraft j and k;
e2: deliverance of taxi plans for aircraft j and k;

4.1.2 Cooperation

4.1.2.1 Need for Cooperation

The need for cooperation of taxi planning for different aircraft arises if planning necessity intervals overlap (fig. 4). As a result of the complexity of the planning problem, it cannot be ensured that taxi plans will be found in time through common planning. Therefore a decomposition of the planning problem is necessary, and can be done naturally by a distributed planning of several planning units which plan independently for just one single aircraft. As aircraft compete for the resource "space" a cooperation is compelled.

4.1.2.2 Design of Cooperation

Following the explanations in the previous sections, the conditions of the design task could be summarized as shown in table 2.

Aspect	Attribute
type of design task	ability of cooperation is already requirement of first unit design
cause of interdependence / cooperation	share of common resources
degree of the need of cooperation	cooperation is compelled to adapt plans to constraints (although planning units solve their tasks independently)
required quality of cooperation	no optimum (overall) solution required
organizational structure	planning tasks are of equal importance
basic architecture	supervised cooperation

Tab. 2: *Description of the design task*

Because of the following reasons the use of a supervisor unit is recommendable:

- ❐ One controller (or a team of controllers) is responsible for overall taxi planning. Therefore a common interface is required to have an influence on planning.
- ❐ There is only a temporary assignment of aircraft to planning units, thus an "authority" is required for task control.
- ❐ Beside cooperation among the distributed taxi planner a coordination of distributed plan monitoring units (necessary to cope with uncertainty and system dynamics) has to take place (see fig.5).

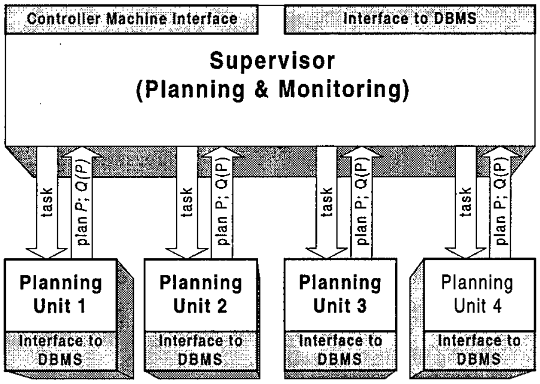


Fig. 5: *Architecture for Supervised Cooperation of Taxi Planning. Since the number of aircraft involved varies some units (in fig. unit 4) may be idle for a certain time.*

4.1.2.3 Cooperation Method: Dynamic Evaluation of Planning Priority

In order to explain the cooperation method of the supervisor more detailed considerations of taxi planning based on a topological airport model are necessary. As taxi planning is reduced to a time-constrained path search on the graph G , the set of possible Solutions Π is discrete and finite if an upper bound for maximum taxi duration is given.

What solutions are possible depends on the set of constraints, therefore $\Pi=\Pi(C)$. Moreover, adding an extra set of constraints to C, especially adding a taxi plan of another aircraft will result in

$$\Pi(C \cup P) \subseteq \Pi(C) \qquad (4\text{–}1)$$

Let be $\Re_Q$ the space of the planning functions of n distributed taxi planner, $P_{k\bullet} \in \Pi_k$ a certain plan of unit k, and

$$\mathbf{Q} = \left[Q_1(P_{1\bullet}),\cdots,Q_n(P_{n\bullet})\right], n \geq 2 \qquad (4\text{–}2)$$

a certain realization of taxi plans of n involved taxi planners then this point shall be termed as a conflict point if there exists at least one pair of plans which are mutual in conflict, i.e. a certain resource is planned to be used by both aircraft (fig. 6). In accordance with this definition a set

$$P_n = \left\{P_j,\ldots,P_k\right\}, j,\ldots,k \in \left\{1,\ldots,n\right\} \qquad (4\text{–}3)$$

is termed a conflict-free set of plans, if n=1 or n>1 and there is no conflict between every pair P_l, P_m *from* $\boldsymbol{P_n}$.

It should be noticed that the solutions Q_D, Q_E , from fig. 6 which might be favored with respect to a cost function Q of overriding importance, could be obtained only if

- ❐ the planning units would calculate not only the best plan, but also the second, third etc., and
- ❐ many combinations of plans have to be proved to be a conflict-free set of plans.

Moreover, the example of fig. 6 shows that different order of precedence of the units may result in substantial different qualities of solutions (Q_B, Q_C).[1]

[1] This example makes also clear, that overlapping intervals of planning necessity times may be desirable, as they give a chance of substantial improvement of

The basic idea of this cooperation method is not to plan accordingly a given (predefined) rank order of aircraft respectively planning units (like "first-known-first-planned"), but to determine a best order of precedence for planning. A more detailed description can be given as follows:

Let

$$\boldsymbol{P}_n^2(\boldsymbol{P}_n) = \begin{Bmatrix} \{P_1\},\ldots,\{P_n\},\{P_1,P_2\},\ldots,\{P_1,P_n\}, \\ \ldots\{P_1,\ldots,P_n\} \end{Bmatrix} = \left\{S_1,\ldots,S_{n^2-1}\right\} \qquad (4\text{–}4)$$

the power set of $\boldsymbol{P}_n$ and

$$\underline{\boldsymbol{P}}_n^2 = \left\{S_i \in \boldsymbol{P}_n^2 \mid S_i \;\; is \;\; conflict-free\right\} = \left\{\underline{S}_1,\ldots,\underline{S}_k\right\},\; k \le n^2-1 \qquad (4\text{–}5)$$

a subset of all conflict-free S_i. Furthermore an evaluation function $q(\underline{S})$ for the $\underline{S}_j$ is needed which should consider (in a suitable way for optimization) at least

❒ the number of plans $|\underline{S}|$ in $\underline{S}$
❒ the quality of the plans included

and might further consider

❒ user or predefined priorities of units (aircraft) given by values of the tuning parameter λ
❒ remaining times to the ends of necessity intervals
❒ properties of Π, like card(Π)>n$_x$, etc.
❒ and other aspects, if required.

A simple example for a useful $q(\underline{S})$ is

$$q(\underline{S}) = \frac{1}{|\underline{S}|^{\alpha}} \sum_{\forall j: P_{j,\bullet} \in \underline{S}} \lambda_j Q_j\left(P_{j,\bullet}\right) \qquad (4\text{–}6)$$
$$\alpha \ge 1, \lambda_j > 0$$

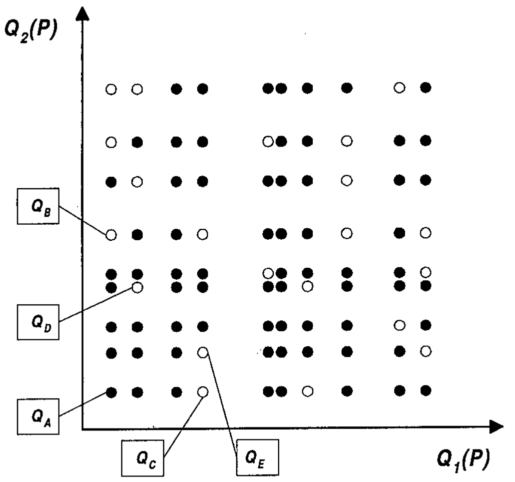


Fig. 6: *Set of Possible Solutions on Example of two Taxi Planning Units. (black: conflict points)*

$\mathbf{Q}_A = \left[Q_1(P_{1,1}),Q_2(P_{2,1})\right] = \left[Q_1^*,Q_2^*\right]$*: Solution with optimum plans of both units cannot be realized because it is a conflict point.*[1]

$\mathbf{Q}_B = \left[Q_1(P_{1,1}),Q_2(P_{2,6})\right] = \left[Q_1^*,Q_2(P_{2,6})\right]$*: Solution which would result if unit 1 would plan before unit 2* $\left(\Pi_1 = \Pi_1(C),\; \Pi_2 = \Pi_2(C \cup P_{1,1})\right)$

$\mathbf{Q}_C = \left[Q_1(P_{1,4}),Q_2(P_{2,1})\right] = \left[Q_1(P_{1,4}),Q_2^*\right]$*: Solution which would result if unit 2 would plan before unit 1* $\left(\Pi_2 = \Pi_2(C),\; \Pi_1 = \Pi_1(C \cup P_{2,1})\right)$

QD, QE possible solutions with respect to a cost function of overriding importance (equation 3.2)

With the help of $q(\underline{S})$ the coordination algorithm can be described as follows:

1. Given all optimum plans from the n involved units $\boldsymbol{P}=\{P_1^*,\ldots,P_n^*\}$ and evaluations $\mathbf{Q}^*=[Q_1^*,\ldots,Q_n^*]$
2. Calculate $\underline{\boldsymbol{P}}_n^2(\boldsymbol{P})$

solution's quality in comparison with a solution of a "first-known-first-planned strategy".

[1] In practice it often happens that taxi plans are not interdependent because they are separated by space or time.

3. Select the best conflict-free set of plans, e.g. an $\underline{S}_j$ from $\underline{P}_n^2(P)$ according to

4. $$S_j = \arg \min_{S_j \in \underline{P}_n^2(P)} \left\{ q\left(S_j\right) \right\} \qquad (4\text{–}7)$$

5. $P \leftarrow P - S_j, \; C \leftarrow C \cup S_j$

6. If $|P|=0$ stop, otherwise cause for all remaining aircraft (according to P) a new planning under consideration of the updated set of constraints C (equation 4-8) and restart with 1 after receiving of plans and evaluations.

4.2 Distributed, Cooperative Planning of Runway Occupancies for Arrivals and Departures

4.2.1 Planning

4.2.1.1 Arrival Planning

Within the last decade several activities were launched that bring computer systems in operation, which support the arrival management and especially the planning task. With respect to the aimed increase of airport capacity, an optimum schedule of arrivals on final approach will be achieved in the end. On final approach, which is usually under control of the tower, there are only minor possibilities to influence the landing time, so that planned schedule of aircraft on final approach will result in certain runway occupancies.

Arriving aircraft have to be separated according (minimum) wake vortex separations, which are depend on the weight categories (h: heavy; m: medium; l: light) of every pair of aircraft succeeding one another immediately. Usually is can be expressed by a separation matrix

$$\mathbf{W} = \begin{bmatrix} w_{hh} & w_{hm} & w_{hl} \\ w_{mh} & w_{mm} & w_{ml} \\ w_{lh} & w_{lm} & w_{ll} \end{bmatrix} \qquad (4\text{–}1)$$

Beside these hard constraints there are further constraints in form of predicted time windows within an aircraft has to reach a certain point (e.g. a metering fix, final approach fix etc.), which are regarded as hard constraints, too, since earliest times cannot be shortened under for technical reasons whereas latest times should be exceeded only as exceptions, because this might lead to safety critical situations.

4.2.1.2 Runway Occupancy Planning for Departures

Runway occupancy planning [14] basically should calculate optimal takeoff times (respectively runway occupancies for takeoff-run), which meet both: the requirements that result from flight plans and from superior traffic planning. So it has to solve a scheduling problem, too. However, in addition to wake vortex separations, minimum separations between departures using the same Standard Instrument Departure Route (SID) also have to be taken into account, even they do not follow directly. Further more there are several constraints for takeoff times - an violation of latest times might be not acceptable or desirable with regard to operational rules, but is not crucial a-priori.

From viewpoint of the operational procedure the main difference to arrival planing is, that the use of a runway by departures is subordinated to the use by arrivals. Therefore occupancies of arrivals are hard constraints to runway occupancy planning. In case that no automatic arrival planning exists, a landing prediction unit is necessary, else plans of the arrival planning units can be used for constraint calculation.

4.2.2 Cooperation

The need for cooperation arises if the runway is used as departure as well as arrival runway at the same time (mixed operation mode).

Usually traffic demand for arrivals as well as for departures is time variant and does not peak simultaneously. As a consequence of that fluctuation the need to stagger arrivals as dense as possible is not really categorical all the time. Especially when departure traffic is high, wider gaps between arriving aircraft would help to release departures more punctual according to their scheduled times.

The disadvantage that some aircraft will arrive later (will later land) then possible will be overcompensated by the fact that in case of an improved departure traffic flow parking positions will be blocked less.

4.2.2.1 Design of Cooperation

The design tasks will be considered under the assumption that an arrival planning unit is always in operation and therefore only minor system modifications will be acceptable.

However, it is assumed that

- □ the set of (static[1]) constraints, especially for that application the separation matrix **W** (eq. 4.7) and
- □ the value of the planning function $Q_A(P_A)$

are accessible.

Further aspects of the design task are summarized in table 3.

Aspect	Attribute
type of design task	making cooperation between existing planning units
cause of interdependence / cooperation	share of common resources
degree of the need of cooperation	cooperation is voluntary
required quality of cooperation	no optimum (overall) solution required
organizational structure	subordinated runway occupancy planning
basic architecture	supervised cooperation

Tab. 3: *Description of the design task*

4.2.2.2 Cooperation Method: Supervised Constraint Adaptation

Let us start the explanation of the cooperation method with a formal description of both planning tasks. Optimum schedules (plans P^*) for arrivals are determined according to

$$P_A^* = \arg \min_{P_A \in \Pi_A(C_A)} \{Q_A(P_A)\} \qquad \textbf{(4-9)}$$

As occupancy planning for departure is subordinated to arrival planning constraints depend on planned (and actual) landings, so

$$C_D = C_D(P_A^*) \qquad \textbf{(4-10)}$$

and therefore

$$P_D^* = \arg \min_{P_D \in \Pi_D(C_D(P_A^*))} \{Q_D(P_D)\} \qquad \textbf{(4-11)}$$

holds.

Now a constraint adaptation function

$$C' = f(C, \mathbf{s}, S), \quad \mathbf{s} \in S \qquad \textbf{(4-12)}$$

should be introduced which is required to fulfill the following condition

$$P \in \Pi(f(C, \mathbf{s}, S)) \Rightarrow P \in \Pi(C) \qquad \textbf{(4-13)}$$

respectively the equivalent condition[2]

$$\Pi(C'(\mathbf{s})) \subseteq \Pi(C) \qquad \textbf{(4-14)}$$

for all values of the parameter vector $\mathbf{s} \in S$:

Now both optimum plans depend on **s**, $P_A{}^*$ directly

$$P_A^*(\mathbf{s}) = \arg \min_{P_A \in \Pi_A(C_A(\mathbf{s}))} \{Q_A(P_A)\} \qquad \textbf{(4-15)}$$

or indirectly, because

$$P_D^*(P_A^*(\mathbf{s})) = \arg \min_{P_D \in \Pi_D(C_D(P_A^*(\mathbf{s})))} \{Q_D(P_D)\} \qquad \textbf{(4-16)}$$

Finally a coordinator function $Q_C(Q_A^*(\mathbf{s}), Q_B^*(\mathbf{s}))$ which is monotonous with respect to Q_A and Q_B is needed so that an optimum parameter $\mathbf{s}^*$ can be determined according to

$$\begin{aligned} \mathbf{s}^* &= \arg \min_{\mathbf{s} \in S} \{Q_C(Q_A^*(\mathbf{s}), Q_B^*(\mathbf{s}))\} \\ &= \arg \min_{\mathbf{s} \in S} \{Q_C(Q_A(P_A^*(\mathbf{s})), Q_B(P_B^*(P_A^*(\mathbf{s}))))\} \end{aligned} \qquad \textbf{(4-17)}$$

In order to avoid that the "sensitivity" of the coordination function depends on the values of the planning criteria a normalized function is used:

$$\begin{aligned} Q_C^* &= Q_C(\mathbf{s}^*) \\ &= \min_{\mathbf{s} \in S} \left\{ \begin{array}{l} \lambda \dfrac{(Q_D^{**} - Q_D^*(P_A^*(\mathbf{s})))^2}{Q_D^{**}} \\ +(1-\lambda) \dfrac{(Q_A^{**} - Q_A^*(\mathbf{s}))^2}{Q_A^{**}} \end{array} \right\} \end{aligned} \qquad \textbf{(4-18)}$$

where $\lambda \in [0,1]$ is a tuning parameter and

$$Q_A^{**} = \min_{\mathbf{s} \in S} \left\{ \min_{P_A \in \Pi_A(C_A(\mathbf{s}))} \{Q_A(P_A)\} \right\} \qquad \textbf{(4-19)}$$

respectively

$$Q_D^{**} = \min_{\mathbf{s} \in S} \left\{ \min_{P_D \in \Pi_D(C_D(P_A^*(\mathbf{s})))} \{Q_D(P_D)\} \right\} \qquad \textbf{(4-20)}$$

[1] constraints which do not depend on system state

[2] with suppressed variation set S

are the values of optimum solutions, that would be reached if the respective unit would control s exclusively.

When defining specific functions for constraint adaptation special conditions of the application domain have to be considered additionally. Here constraint adaptation is done by transforming the matrix **W** to **W'** using a very simple function with a one-dimensional parameter s (equation 4-21, 4-22)

$$\mathbf{W}' = s\mathbf{W},\ \ s \in [1,3] \qquad \textbf{(4-21)}$$

or alternatively

$$\mathbf{W}' = \begin{bmatrix} sw_{hh} & sw_{hm} & w_{hl} \\ sw_{mh} & sw_{mm} & w_{ml} \\ w_{lh} & w_{lm} & w_{ll} \end{bmatrix},\ s \in [1,3] \qquad \textbf{(4-22)}$$

Against the background of time-stressed decision demand planning has to be done quickly. For that reason S has to be a discrete set with only a few members. As planning tasks for different s can be solved simultaneously the number of members can be adapted to the number of CPU's available.

5 Summary and Further Investigations

As shown on the example of APTM cooperation among distributed planning units is either an absolute need or a means to improve the overall management. Although the development of planning units, which support local management tasks, is currently still a field of research, the cooperation task has to be brought into the focus of interest when operational concepts for the next decade (at least), like the ASMGCS concept, are outlined.

For that reason the making of cooperation will be considered mainly under two aspects:

- ❒ There is an overall design of a set of planning units which will be put commonly into operation.
- ❒ The planning units involved, which should be made cooperative, are (all or partly) already in operation.

With regard to both aspects first results of research were described on examples. In the paper especially the static case was treated, e.g. the situation at a certain time. However, in a (highly) dynamic domain, where uncertainty of information, unknown future events and the human behavior, prevent a exact prediction of the future system behavior, plans have to be adapted constantly. Thus the sequence of cooperative plans, which may be implemented only partly, affects the quality of the overall management. Furthermore, a certain plan stability, e.g. similarity between consecutive plans, is required, because humans are involved as well in management as in plan implementation. Since up to now only a few and very specific methods are available to determine the dynamic behavior of cooperation analytically (e.g. [15][16]), investigations will be done with the help of simulations which include human controllers as well.

The interaction between machine and the human operator, who remains responsible for management, and therefore is allowed to influence planning by setting constraints, by modifying planning and cooperation functions, as well as the one who can accept, change, or refuse plans will be one area of further work.

6 Abbreviations

- ❒ ATM: Air Traffic Management
- ❒ APTM: Airport Traffic Management
- ❒ TARMAC: Taxi And Ramp Management and Control
- ❒ DLR: German abbreviation for: "Deutsches Zentrum für Luft- und Raumfahrt"
- ❒ ASMGCS: Advanced Surface Movement Guidance and Control System
- ❒ TMA: Terminal Maneuvering Area
- ❒ DBMS: Database Management System
- ❒ HMI: Human Machine Interface
- ❒ SID: Standard Instrument Departure Route
- ❒ CPU: Central Processor Unit

7 Literature

1) ICAO: Manual of Advanced Surface Movement Guidance And Control Systems (A-Smgcs) Regional Provisions (Eur) Eur_Draft_01, 03.11.97, Paris, 3-5 November 1997, Special AOPG meeting on A-SMGCS, Brussels
2) U. Völckers, D. Böhme: Dynamic Control of Ground Movements: State-of-the-Art Review and Perspectives, AGARD-R-825, 1997
3) N. V. Findler, R. Lo: An Examination of Distributed Planning in the World of Air Traffic Control. Journal of Parallel and Distributed Computing. Vol. 3, 1986, pp. 411-431
4) F. V. Schick, U. Völckers: The COMPAS System in the ATC Environment. DLR-Mitt. 91-08 (1991)
5) R. Wall, W. Cook, J. DeArmon, E. Beaton: Self-Managed Arrival Resequencing Tool (SMART): An Experiment in Collaborative Air Traffic Management. Journal of ATC, April-June 1997

6) DEFAMM: Demonstration Facilities for Airport Movement Management. EC FP IV DGVII: Air Transport Project, CT Number: AC-95-SC.302, 1996

7) K. Klein, G. Mansfeld: Projektplan Rollverkehrsmanagement (TARMAC), DLR (German Aerospace Center), IB 112-97/33, 1997

8) EUROCAE: Surface Movement Guidance and Control Systems. Interim report of Eourocae Working Group 41, ED-200, June 1993, Paris

9) D. Böhme: Improved Airport Surface Traffic Management by Planning: Problems, Concepts and a Solution — TARMAC. H. Winter (Ed.): Advanced Technologies for Air Traffic Flow Management. DLR-Seminar Series, Bonn, Germany, April 1994. in Lecture Notes in Control and Information Sciences, Springer Verlag, ISBN 3540198954

10) U. Völckers, U. Brokof, D. Dippe, M. Schubert: Contribution of DLR to Air Traffic Enhancement within the Terminal Aria. AGARD, 56th Conference on GCP, Paper No. 10, Berlin, 1993

11) N. V. Findler, G. D. Elder: Multi-Agent Coordination and Cooperation in a Dynamic Environment with Limited Resources. Artificial Intelligence in Engineering, 9, pp. 229-238, 1995

12) E. H. Durfee, V. R. Lesser: Incremental Planning to Control a Blackboard Based Problem Solver. Proceedings of AAAI, Pittsburg, PA, August 1986, pp. 58-64.

13) D. D. Corkill, V. R. Lesser: The Use of Meta-Level Control for Coordination in Distributed Problem Solving Network. Proceedings of the Eighth IJCAI Conf., Karlsruhe, pp. 748-756, 1983

14) D. Böhme: Airport Capacity Enhancement by Planning of Optimal Runway Occupancies. Proceedings of the "39. Internationales Wissenschaftliches Kolloquium", Technische Universität Ilmenau (Thür.), Bd. 3, S.250-256, 1994, ISSN 0943-7207

15) M. B. Zaremba, K. J. Jedrzejek, Z. A. Banaszak: Design of Steady-State Behavior of Concurrent Repetitive Processes: An Algebraic Approach. IEEE Trans. on Systems, Man, and Cybernetics – Part A: Systems and Humans, Vol. 28, No. 2, March 1998, pp.199-211

16) B. Gaujal: Optimal allocation sequences of two processes sharing a resource. Tech. Report 2223, INRIA, France 1994.

Optimal Decision-Making and Battle Management

by

Dionyssios A. Trivizas, Ph.D
5 Lycabetou Street
10672 Athens
Greece

1. INTRODUCTION

In the era of "Star Wars", modern war-machines have become extremely complex in terms of automation and sophisticated technologies. The sophisticated battle equipment needs to be effectively coordinated so as to achieve maximum strike-power with minimum losses and this is the point where Operations Research and Optimal Decision-Making under uncertainty come into play.

This paper presents the concepts involved in optimal real-time decision-making and gives an example of a military application in Battle Management (BM).

Battle management has a strategic planing component and a tactical component, i.e. real-time monitor and control. Mathematical modeling using the vast computational power of present day computers may carry out both these complex tasks effectively. Sensor data and other relevant intelligence about enemy deployment and intentions combined with knowledge of our own system can be utilized in order to:

- Make cost effective provisions in the planning phase
- Make the most effective real (battle)-time use of personnel and ammunition.

After setting the stage by describing a realistic battle scenario, we introduce the concept of the decision tree, which forms the fine grain representation of the solution space. Typically, such a tree has to be evaluated in a backwards from the future fashion. We also discuss the notions of static and dynamic problems in a synthesis leading up to a practical algorithm for real time battle management.

2. THE SCENARIO

The scenario described here comes from a space based battle management system whereby:

1. Invading nuclear warheads, the "targets", from the (ex) Soviet Union targeting cities and military bases in the US, have just been placed in ballistic orbits over the North Pole, which will last for about 2000 seconds.
2. Space-based sensors perform multi-sensor tracking, by fusing their data, in order to assess the invading warhead (target) orbits. Guessing these orbits provides the basis for assigning values to the invading warheads whereby a small town is worth less than a big City.
3. Space-based satellite "weapon-farms" referred to as "weapons", carrying interceptors (shots), are moving on known controlled meridian orbits and come into shooting range of invading warheads. Typical interceptor-fly out times are about 200 seconds.
4. For each weapon-target pair there is a "time window of opportunity", during which there is a finite probability of an interceptor fired from weapon "i" to hit and destroy target "j", known as "probability of kill"
5. Spaced-based "Battle Manager" computers direct the battle.

3. THE OBJECTIVE

The Objective of the battle management system is to make the most effective use of the available resources (interceptors in this case) in order to protect the country. However, destroying as many invading targets as possible is not enough. As the value of every warhead depends on its destination, one has to:

> "Minimize the sum of target values
> that "leaks" through the defenses".

The complexity of the problem is further increased by the variable battle geometry whereby time windows of opportunity are randomly dispersed over time.

4. THE STATIC BATTLE – THE AUCTION WEAPON-TARGET ASSIGNMENT (WTA) ALGORITHM

A battle is a real-time dynamic game, whereby the opponents make moves in succession assessing all along the results of their strikes as well as the opponent's remaining strength and intentions.

Why then does one need to consider static **Weapon-Target Assignment?** One answer, borrowing an analogy from adiabatic compression in Fluid Mechanics, is that

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

dynamic is but a succession of static phases. In this sense static is a time-wise reduction of the dynamic problem.

To help crystallize the concept of "static", consider the very simple case where a hunter has two shots and the opportunity to hit two rabbits A and B where the value of A is larger than the value of B. Under the assumption that the hunter may only shoot once, the problem is static. The solution will determine whether the hunter hits twice the fat rabbit A, or he hits skinnier B as well.

5. THE DYNAMIC BATTLE: SHOOT-LOOK-SHOOT STRATEGY

Continuing the example, suppose the hunter had the opportunity to assess the result of his first shot and shoot again. Clearly, this presents him a better opportunity. If the hunter misses the fat rabbit A with the first shot, he will shoot at it again, instead of shooting rabbit B.

One may prove mathematically by assigning values to A, B and to the probability of kill that this strategy has higher expected return.

This elementary "dynamic" example illustrates the succession of two trivial static problems in a "shoot – look (assess) – shoot" fashion.

In general, static problems are instant, deterministic and simple, amenable therefore to computation (tame). They are therefore candidates as means (subroutines) of solving dynamic problems, that are hard since they involve:

- Enumeration of all possible outcomes of one's actions, including enemy reactions. Therefore, dynamic problems are probabilistic or "stochastic" decision making problems.

- Time evolution

- Economy of resources spread over time.

6. THE DYNAMIC DECISION MAKING PROCESS: THE DECISION TREE

Typically, in such complex dynamic problems one works backwards from the future. To help understand the process suppose that one listed all battle options in the form of a decision tree whereby decision branching nodes interleave with opponent reaction nodes.

Figure 1 shows schematically the form of a decision tree. The arcs (arrows) branching out of each decision node lead to enemy reaction nodes and vice versa.

The leaves of such tree, are the outcomes. There we assess the gain or damage to what we are trying to protect. Assigning values to such leaves is a mechanism of prioritizing the battle outcomes so as to be able to opt for the optimal.

Deciding what to hit, in the course of a battle, corresponds to moving on a path, from the root towards some leaf of the battle's decision tree abstraction.

Typically, outcomes may be equivalent in terms of their value. Therefore, it makes sense to define battle objectives as sets of equivalent outcomes. This is a principle of aggregation that reduces computational requirements as will be shown below.

6.1 Probabilistic Evaluation - Expected Values of Reaction Nodes

So far in our decision tree we have missed:

- The probabilities of success of our actions, associated with branches out of decision nodes. In the space based scenario they depend on the assessment of invading target orbits and the evolving 3-D geometry.

- The probabilities of choice and success of enemy reactions, associated with branches out of reaction nodes. In our scenario, the enemy uses decoys to fool our sensing ability. In general, however, we have to list enemy reactions and assign them a probability value.

  These probabilities are the "essence" of our raw intelligence, be it spies, sensors or collective experience.

- The probability of a final "leaf" outcome, is the product of probabilities associated with the tree-branches forming a path that leads from the present decision node to the specific leaf.

Notice that, as the battle progresses, we move deeper into the tree and the leaf-outcome probabilities change.

Using the path probabilities one may compute for every decision branch of the tree, i.e. for every reaction (enemy-decision) node, an "expected value" which is the sum of leaf-outcomes weighted by the respective path probabilities.

Back from the Anticipated Future

The computed reaction-node values, which guide our spot-decisions during the battle, have to be computed working backwards from the leaves, i.e. "the anticipated future". This "backwards from the future" approach is typical of stochastic decision making problems and we will demonstrate its use as a "design philosophy" in coming up with practical solutions to our space-based battle scenario.

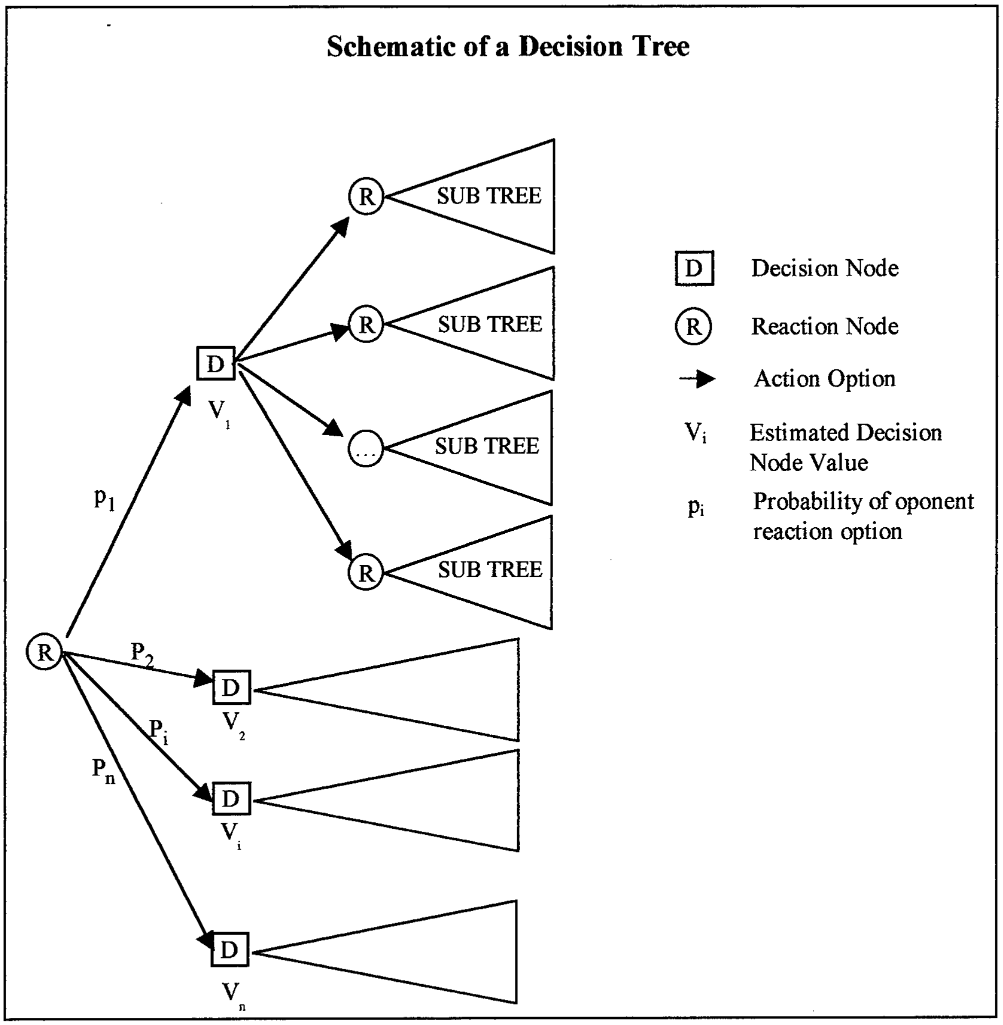

Schematic of a Decision Tree
SUB TREE
SUB TREE
SUB TREE
SUB TREE
R
R
R
R
D
D
D
D
V1
V2
Vi
Vn
p1
P2
Pi
Pn
D Decision Node
R Reaction Node
Action Option
Vi Estimated Decision Node Value
pi Probability of oponent reaction option

Figure 1: Schematic Illustration of the Decision Tree

6.2 Complexity

The decision tree is the central concept of incremental, i.e. real-time decision-making. It represents the finest grain solution space, which contains the sequence of optimal tactical decisions. The optimal decision sequence may only be found by enumeration of all possible tree outcomes and computing node values.

Calculating, however, a node value in the decision tree is a laborious process of shear magnitude that grows exponentially with the size of the problem (in our scenario determined by number of weapons and number of targets). In particular it requires:

- Amounts of computational resources, such as memory and computation time, which grow exponentially with problem size
- Efficient memory coding, for saving intermediate results, to avoid duplication of calculation.

6.3 Practicality – Heuristic Approaches

Optimal solutions to problems are only good if they are available in time. Therefore, in practice one often resolves in "heuristic" solutions based on the intuition of the problem's analysis and experience. Typically, heuristics involve:

- Aggregation of parts of the decision tree whereby one makes quick estimates of node-values. One type of such aggregation is merging of intermediate nodes when they are equivalent in terms of friend and foe states.
- Time discretization, whereby battle-time is split in phases. This may be viewed as a time-wise aggregation.
- Pruning of the tree using feasibility arguments and order of magnitude analysis

- Taking shortcuts such as using fast greedy weapon to target assignment approaches that compromise local optimality.

7. AN EXAMPLE OF DYNAMIC BATTLE MANAGEMENT: "THE ANTICIPATING SPACE BASED WEAPON TO TARGET ASSIGNMENT ALGORITHM (AWTA)"

Having described the nature of Dynamic real-time problems, we present the **"The Anticipating Space Based Weapon to Target Assignment Algorithm (AWTA)"** which is an application of the principles laid out above.

The AWTA splits battle time in two phases or periods and uses a static algorithm as a subroutine. It was conceived by author, while working at ALPATECH Inc., Boston Mass., US, and it was called the Anticipating Closed Loop (ACL) algorithm. The ACL computer code was parallelized and run on hyper-cube computer architectures at the Rome Air defense Center and the Argon National Labs produced dramatic improvements over other approaches.

The static problem was approached in two ways:

- Using a greedy heuristic algorithm, whereby weapon to target assignments are made following A Maximum Marginal Return (MMR) principle

- Using the auction algorithm, an optimal solution method using coordinate descent on a convex objective function. This mathematical method has an entertaining as well as intuitive interpretation of an "auction" whereby assignments are made through a pricing and bidding mechanism.

7.1 The Auction Algorithm

In the auction algorithm one may have the targets for instance bid for their favored weapons, or vice versa as follows:

- Bidding is based on "profit" defined as the value of the specific weapon to target assignment minus weapons price. Weapon prices are initially zero.

- In a price determination phase, each weapon orders reachable targets in decreasing profit and raises the price of its first choice so as to make it equivalent to the second best.

- In a bidding phase, we have a coordinated reassignment of weapons to targets.

- Price determination and bidding are repeated until all targets are assigned to weapons.

As bidders can determine prices independent of each other and in parallel, the auction algorithm was considered a candidate for parallel implementation whereby the battle managers would collectively arrive sooner to an optimal solution.

The author was first to implement the auction algorithm achieving a ten-fold performance improvement over the classic graph theory Ford-Folkerson assignment algorithm used thus far. This improvement resulted from noticing and exploiting the solution memory feature contained in the weapon prices, which led to avoiding repetition of calculations.

7.2 The Open Loop Feed-Back Algorithm (OLFB)

Until the conception of the ACL algorithm, the practical algorithm used for the space-based weapon to target assignment problem was a re-planing algorithm, which used greedy MMR or auction static subroutines. At the end of the re-planing interval an assessment was made of surviving targets, i.e. "feed-back", and the static algorithm was repeated with the updated set of surviving targets and the remaining inventory of interceptors.

7.3 The Two-Period Time Split

Trivizas introduced the idea of splitting the battle time in two periods or phases. Choosing the time split just before the reentry of the warheads would allow just enough time for a meaningful static problem in the second period P_2.

Another way of looking at this problem is that ultimately there is going to be a last static shoot in the shoot-look-shoot sequence and we want to induce the best conditions for that ultimate shoot.

The anticipating algorithm (AWTA) starts by solving a static problem in P_2, i.e. works backwards from the future as follows:

I. All targets are admitted in P_2 since one does not know a priori which will survive

II. Interceptor-inventory is roughly divided in the two periods

III. As targets are going to be eliminated in P_1 the number of interceptors in P_2 are virtually inflated so as to maintain a realistic weapon to target ratio

IV. Based on the solution of the static problem in P_2 new "modified target values" are computed for the targets to be used in period 1.

V. Modified values are used to solve a static problem in P_1

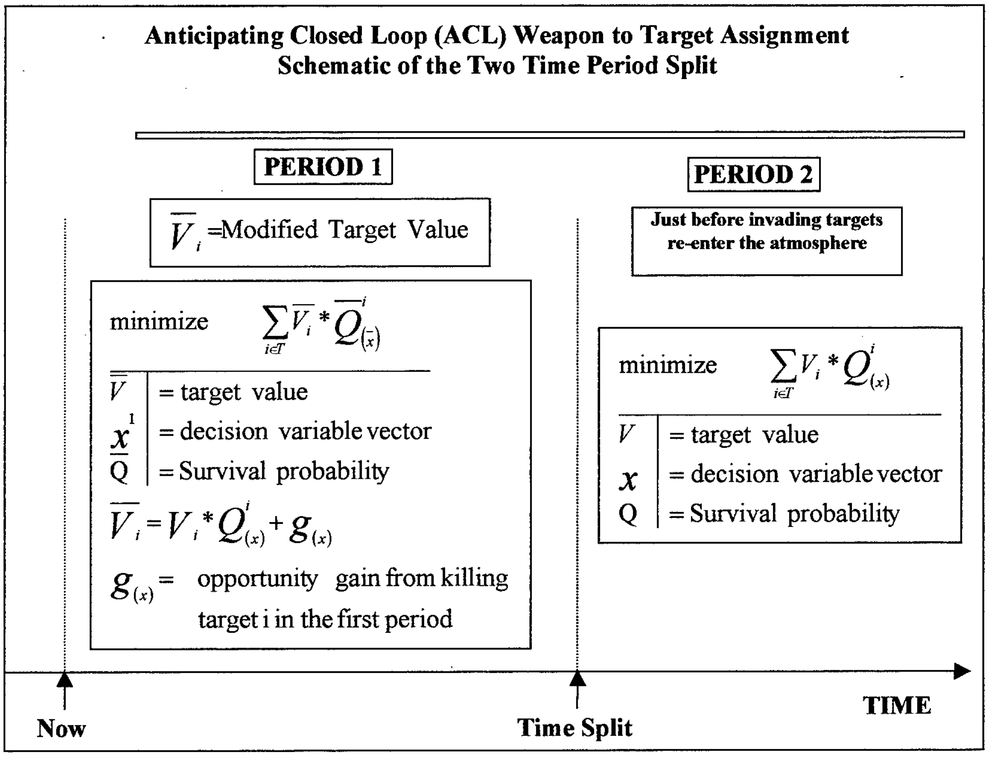

Anticipating Closed Loop (ACL) Weapon to Target Assignment
Schematic of the Two Time Period Split
PERIOD 1
$\overline{V}_i$ =Modified Target Value
minimize $\sum_{i \in T} \overline{V}_i * \overline{Q}^i_{(\bar{x})}$
$\overline{V}$ = target value
$\underline{x}^1$ = decision variable vector
$\overline{Q}$ = Survival probability
$\overline{V}_i = V_i * Q^i_{(x)} + g_{(x)}$
$g_{(x)}$ = opportunity gain from killing target i in the first period
PERIOD 2
Just before invading targets re-enter the atmosphere
minimize $\sum_{i \in T} V_i * Q^i_{(x)}$
V = target value
x = decision variable vector
Q = Survival probability
Now
Time Split
TIME

Figure 2 : Schematic representation of the Anticipating Closed-Loop Algorithm

VI. A sensitivity analysis based on the results so far determines which weapons gain from transferring a shot between the two periods

VII. Following the shot transfer, steps I –VII are repeated until no weapon gains from transferring shots.

7.4 The Modified Target Values

Figure 2 shows the schematically the two time period split and the mathematical formulation of the static objective functions in the two periods.

At first one may notice that destroying a target in P_1:

- eliminates only the target's survival value from period P_2
- frees up the interceptors that this target would absorb and which can be used against remaining targets at a "price" as we called the average opportunity gain, borrowing a term from economics.

Clearly, the modified target value, defined as the sum of period 2 survival value plus opportunity gain is the mechanism used to transfer information from the future (P_2) to the present (P_1).

To appreciate the benefit of this approach consider two targets:

- target ***a***, easily reachable in P_2
- and ***b*** unreachable in P_2.
- with initial value of ***a*** slightly higher the value of ***b***

the modified target value of ***a*** is going to be a lot less than the modified value of ***b***. Thus , target ***b*** will be correctly more vulnerable in period P_1.

The simple re-planing algorithm is incapable of judging the time-wise distribution of opportunities and is very likely to allow high value targets in the second period.

Thus, the AWTA even when it employs the greedy MMR heuristic as a static subroutine produces far better results than the simple re-planing using optimal static solutions.

It was observed in simulated battles, using Monte Carlo random number techniques, that the leakage using the simple re-planing approach decreased with decreasing re-planning interval, the external parameter with regard to the optimization. This monotonic behavior was tested down to a re-planning interval of 50 seconds, which is a quarter of the mean fly out time.

The optimal re-planning interval for the ACL was around 200 – 300 seconds, which is of the order of fly-out times. This makes intuitive sense, since one should allow the interceptors to reach the targets before making an effective assessment.

This observation explains how the ACL achieves reduced leakage with overall less computation. Despite its increased computational complexity increase of a factor of 2 to 4, it only requires on run for every 4 or 5 runs of the simple re-planning algorithm.

This resulted in improved economy of the shot inventory as it allowed for better kill assessment.

7.5 Recursive Multi-Period Time Split

The author experimented further with reapplying recursively the two period split to the first period, and the first sub-period etc. up to a maximum of m periods is achieved. "m" is determined such that the duration of the each sub-sub…period equals the mean interceptor fly out times (approx. 200 sec's).

Also, recursive time-split allows for time-varying probabilities of kill.

7.6 Performance testing.

Testing of these algorithms was done in limited and large-scale battle scenarios on a variety of computers ranging from PC's and VAX mini-computers, to supercomputers with hyper-cube architectures at the Rome Air Defense Center and the Argon National Labs in the US.

The computer simulation contained warhead and weapon orbit data, and used a random number generator to simulate the random kill process. This is known as the Monte Carlo technique.

The results were collected over a large number of simulated battles, typically 100 to 1000, using different random number seed (starting point), so as to have a meaningful sample for computing the mean and range of the leakage produced by the candidate WTA logic.

Under these conditions, the anticipating algorithm was seen to outperform by order of magnitude the simple feedback algorithm in terms of the leakage through the defenses.

Figure 3 shows schematically a performance comparison of the ACL and OLFB algorithms. The observed ACL algorithm mean leakage values were anywhere from one half to one fifth of the respective values using open loop feed-back, the ACL superiority growing with problem size.

The amount of variance in the relative performance is determined by the internal structure of the scenario, i.e. the weapon – target deployment geometry, which dictates the variance in the probability of kill matrix.

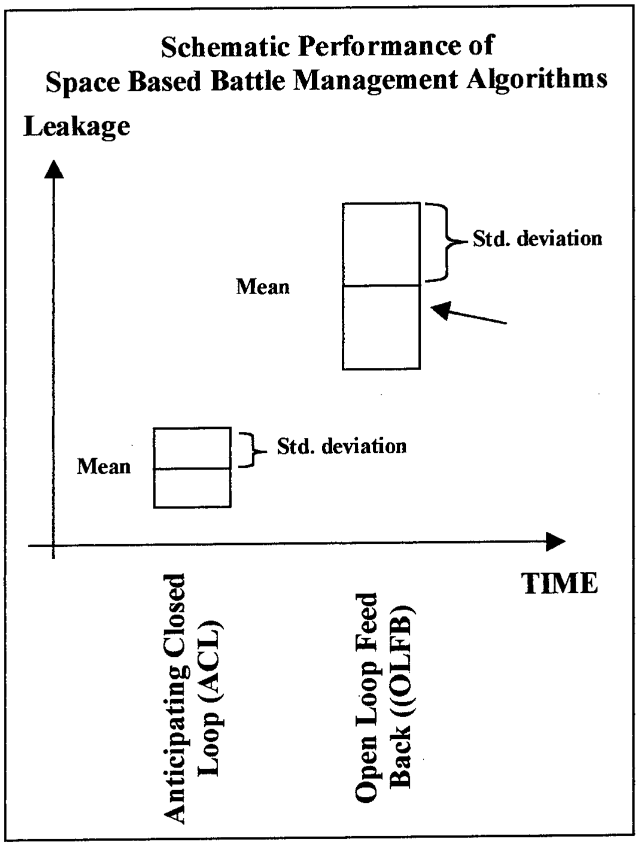


Figure 3: Comparison of results

Unfortunately, due to the defense-related nature of this work computational results are not widely available. They would require special permission in order to access them and anyone interested may contact the author.

8. CONCLUSION

The main purpose of this paper is to present the concept of Dynamic Decision-making using the space-based battle management as an example.

Hopefully, the reader has obtained a computational perspective that improves his intuition on the complexity of the real-time decision making.

We have also described the anticipating algorithm, that performs a time-wise aggregation on the decision tree, which is the fine grain representation of the solution space.

On Vehicle Allocation to Targets in Mission Planning*

Sunil Choenni

Informatics Division,
National Aerospace Laboratory,
P.O. Box 90502, 1006 BM Amsterdam
The Netherlands
email: choenni@nlr.nl

Abstract

A military mission consists of a large number of activities, such as launching weapons to targets, deploying vehicles to targets and pointing out an efficient route for each vehicle, etc. Planning plays an important role in many of these activities. Although the apparently different nature of a plan in each activity, we feel that many of these plans can be captured in one formalism. In this paper, we present such a formalism and exploit it to solve a planning problem. Our formalism is based on the well-known notions of sets and lists. A plan can be regarded as a list of operations on sets of objects. The problem to deploy vehicles to targets with acceptable costs will be expressed in this formalism. To solve this problem, we propose a two step approach. In the first step, we generate lists of targets that are feasible for each vehicle. In the second step, we select for each vehicle at most one list, such that, finally, all objectives for the mission concerning the allocation of vehicles to targets are attained.

1 Introduction

Planning is an essential requisite to fulfill (complex) military missions successfully [1]. The purpose of planning is to generate and maintain efficient plans in order to guide a mission. Important planning activities are, among others, determination of an efficient route for a vehicle and allocation of a set of vehicles to a number of targets. Although the nature of these activities is apparently different, many of these activities can be captured in one formalism. In this paper, we present such a formalism and exploit it to solve a problem in the field of mission planning.

The problem addressed in this paper is the following: *given a set of vehicles, e.g., aircraft, located on geographically different bases, and a number of targets with relevant characteristics; deploy these vehicles to the targets in an efficient way.* In the following we refer to this problem as Vehicle Mission Problem (VMP). While we assume that relevant information with regard to the vehicles and bases is precisely known, we do not make this assumption for the targets. Information with regard to a target is the product of intelligence gathering, and may be updated frequently. As a consequence, a plan should be revised in an efficient way whenever an update occurs.

Since the VMP is NP-complete, there exists no technique that solves the problem in polynomial time. To attack the problem, we have developed a formal model in terms of a cost function, which takes as input a plan and computes its associated cost. We were able to prove that this function is monotonous non-decreasing. We exploit this property in combination with domain knowledge to search efficiently for a solution for VMP, resulting into a two step approach. In the first step, we generate for each vehicle the lists of targets that are appropriate for this vehicle. In the second step, we select for each vehicle at most one list such that, finally, all objectives for the mission are attained with acceptable cost. Theoretically, both steps have an exponential complexity. Therefore, we have derived a number of rules to control this complexity. Furthermore, we present a procedure to update a plan adequately whenever previously unknown information becomes available.

A number of efforts has been reported to solve

*This research has been performed in the scope of the EUCLID RTP6.1 project.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

problems similar to the one we have addressed, see among others [1, 2]. In these efforts, more or less the following approach is followed for this type of problem. A problem is typically formulated as an optimization problem in a "black box" way, i.e., in the form of *optimize a function f(.), which is subjected to a set of constraints.* Then, an algorithm is selected or developed to search for a solution. In [4], these algorithms are categorized as algorithms that guarantee an optimal solution and algorithms that do not guarantee this. The algorithms in the former category are mainly based on (mathematical) classical techniques, such as dynamic programming, gradient methods, etc., while the algorithms of the latter category are mainly based on more novel techniques, such as neural networks, knowledge base technology, etc.

Our approach differs on three main points from above-mentioned approach. First, we have chosen a formalism to express a problem that is better accessible for domain experts than the black box formulation. This allows to add domain knowledge and modifications to a plan easily. For example, in our formalism we can easily express relationships as "Action A should be followed/preceded by action B", while the black box does not explicitly support this kind of relationships. Second, often, a planning problem is divided into two subproblems, namely an allocation and a scheduling problem [3]. In our approach, we do not make this difference since allocation and scheduling may strongly related to each other. Third, we have integrated the strong points of classical and more novel techniques to solve VMP. We have taken advantage of the fact that the cost function for VMP is non-decreasing as well as from heuristics that are used by experts in this field. In most efforts, either a classical or a more novel technique is used to solve the problem.

The remainder of this paper is organized as follows. In Section 2, we describe VMP in more detail, and in Section 3, we derive a mathematical model for the problem. Then, in Section 4, we discuss our approach to select plans and to update plans according to previously unknown information. In Section 5, we show how our approach can be applied to solve a problem in the naval domain. Preliminary results based on this application will be discussed as well. Finally, Section 6 concludes the paper.

2 Problem definition

In this section, we define VMP in more detail. In a military setting, vehicles, e.g., aircraft, are located on geographically different bases. These vehicles should be deployed to targets. The allocation of vehicles to targets and in which order they are deployed to targets are expressed in a plan.

While relevant data with regard to each vehicle are precisely known, this may be not true for targets. Information with regard to a target is the product of continuous intelligence gathering, and may be frequently updated. As a consequence, whenever information with regard to a target is updated, the plan should be updated according to the new information. Our goal is to contribute to an efficient selection and update of plans.

For the time being, we assume that a vehicle can be deployed for at most one list of targets in a plan, i.e., a vehicle appears at most once in a plan. Before elaborating our problem, we define the notion of a plan more precisely. For this purpose, we use the well-known set and list notations. Perhaps unnecessarily, we note that the order of elements in a set is irrelevant and that a list can be regarded as an ordered multiset (thus allowing duplicates).

Definition 1: Let $\mathcal{V} = \{v_1, v_2, v_3, ..., v_m\}$ be a set of vehicles, $\mathcal{T} = \{t_1, t_2, t_3, ..., t_n\}$ a set of (hostile) targets, and $X_i = \{(V_1, \mathbf{T}_1), (V_2, \mathbf{T}_2), (V_3, \mathbf{T}_3), ..., (V_k, \mathbf{T}_k)\}$, in which $V_i \subseteq \mathcal{V}$ and $\forall i, j; i \neq j : V_i \cap V_j = \emptyset, 1 \leq i, j \leq k$, and $\mathbf{T}_i = [t_1, t_2, t_3, ..., t_l]$, $t_l \in \mathcal{T}$, $1 \leq l \leq n$. A plan P is defined as $P = [X_1, X_2, X_3, ..., X_s]; \forall X_p, X_q; p \neq q : x \in X_p, y \in X_q \Rightarrow x.V_i \cap y.V_j = \emptyset$.

In the following, an X_i will be called an *action* set.

Example 1 Let us consider a plan $P = [\{(\{v_1, v_2\}, [t_1, t_2])\}, \{(\{v_3\}, [t_3])\}, \{(\{v_4\}, [t_5, t_6])\}]$. Note that this plan consists of three action sets namely, $X_1 = \{(\{v_1, v_2\}, [t_1, t_2])\}$, $X_2 = \{(\{v_3\}, [t_3])\}$, and $X_3 = \{(\{v_4\}, [t_5, t_6])\}$. According to this plan, v_1 and v_2 (simultaneously) should successively destroy targets t_1 and t_2. Then, v_3 should destroy t_3. Finally, v_4 should destroy targets t_5 and t_6, successively. If it is not necessary that v_4 starts its actions after that v_3 has completed its actions, this can be expressed by taking the union X_2 and X_3 in plan P. So, $P = [\{(\{v_1, v_2\}, [t_1, t_2])\}, \{(\{v_3\}, [t_3]), (\{v_4\}, [t_5, t_6])\}]$. □

Two main factors can be distinguished in the selection of plans namely, the effectiveness of a plan and the cost of a plan. The effectiveness of a plan depends on the extent that a defined goal is met. A plan that totally meets a goal is called a *complete* plan.

We set ourselves as goal to select complete plans with acceptable cost with regard to a given set of targets. Furthermore, whenever information with regard to a target is updated, the selected plan is updated according to the new information.

The cost of a plan is determined by parameters, such as fuel consumption, types and number of vehicles involved in a plan, etc. However, in general, a plan P according to which a target t should be destroyed by vehicle v will be cheaper than any plan P' that includes plan P. This will be used to control the search space of plans. In the following sections, we formalize this observation and exploit it in searching for plans.

3 Mathematical model

In this section, we define a mathematical model for the problem introduced in the previous section, and prove some properties for this model. On the basis of these properties, we introduce, in the next section, a solution for the problem.

As discussed in the foregoing, our goal is to select complete plans with acceptable cost. In order to fulfil this task, we define a cost function to compare different plans. The cost function is defined in such a way that incomplete plans get ∞ as cost value. Before defining the cost to move to a target t by vehicle v, we define under which conditions a target can be attained by a vehicle.

Definition 2: Let $\mathcal{B} = \{b_1, b_2, b_3, ..., b_m\}$ be a set of bases. Then, a list of targets $\mathbf{T} = [t_1, t_2, t_3, ..., t_n]$ is *attainable* by a vehicle v_j, which is located at a base $b_j, j \leq m$, iff

1. v_j has the capacity to bridge the distance between b_j and all targets of $\mathbf{T}$ in the given sequence (without any stopovers), and afterwards
2. There exists a $b_h \in \mathcal{B}$, such that $b_h = min_{b_l \in \mathcal{B}} C_{\text{base}}(t_n, b_l)$, in which $C_{\text{base}}(t, b) \geq 0$ represents the *minimal* cost that is required by a vehicle to move from target t to base b, and v_j has the capacity to bridge the distance between t_n and b_h.

Note, b_h is the base that can be reached with the lowest cost by a vehicle from the last target in a target list.

The cost (which assumes values ≥ 0) involved in attaining a list of target T by a vehicle v starting from a base b is defined as follows:

$$C_{\text{att}}(v, b, \mathbf{T}) = \begin{cases} f_{v,b,\mathbf{T}} & \text{if } \mathbf{T} \text{ is attainable by } v \\ \infty & \text{otherwise} \end{cases}$$

Once a vehicle attains a target, some actions should be taken to destroy the targets. This cost depends on the actions and weapons used for these actions. If a vehicle is unable to perform these actions, e.g., it is not able to carry the suitable weapons, the cost is defined as ∞. Let us define the cost (assuming values ≥ 0) involved with performing actions on a target t by a vehicle v as follows:

$$C_{\text{action}}(v, t) = \begin{cases} c_{v,t} & \text{if } v \text{ is able to perform the} \\ & \text{defined actions above } t \\ \infty & \text{otherwise} \end{cases}$$

The general cost function for a plan P is defined as:

$$C(P) = \sum_{X_i \in P} \sum_{x \in X_i} (\sum_{v_k \in x.V} (C_{\text{att}}(v_k, b_k, \mathbf{T}_i) + \sum_{t_l \in \mathbf{T}_i} C_{\text{action}}(v_k, t_l)))$$

Before showing that function C is monotonous non-decreasing, we note the following with regard to this function. The cost of a plan does not depend on the order in which the action sets of a plan are performed. We feel that the order of action sets becomes significant if a vehicle is allowed in more than one action set of a plan. In that case a vehicle has various alternatives with probably different cost to perform its action sets. Since we have assumed that a vehicle appears at most once in a plan, we feel that the order of action sets can be neglected. However, the extension of our cost function, such that it is able to take the order of action sets into account, is straightforward. For example, one may introduce a penalty function for two action sets that are not in a proper order in a plan. A penalty function C_{pen} takes two action sets X_i and X_j of a plan as input, and produces a penalty in terms of cost if they are in an inappropriate order, i.e., if X_i precedes X_j while the reverse is desired, and zero otherwise. Let the penalty due to undesired orders on a plan be $S(P) = \sum_{X_i \in P} \sum_{X_j \in P, i \neq j} C_{\text{pen}}(X_i, X_j)$. Then, function C can be extended to a function $\tilde{C}$ that takes the order of action sets into account as follows: $\tilde{C}(P) = C(P) + S(P)$.

Although a plan can be regarded as a set of action sets instead of a list since we consider function C, we still regard a plan as a list. As illustrated above, function C can be easily extended such that the order of action sets has its impact on the cost of a plan. Therefore, we anticipate on this situation and regard a plan as a list of action sets in solving VMP. Note that regarding a plan as a set of action sets simplifies VMP.

Let us return to function C. In the following, we prove some properties for this function.

Proposition 1: The cost function C is monotonous non-decreasing.

Proof: We prove consecutively that C_{att} and C_{action} are monotonous non-decreasing. Let $\mathbf{T} = [t_1, t_2, ..., t_i]$ be a list of targets attainable by a vehicle v that starts from a base b, and b_j^k the cost to move from target t_k to the j-th base. The cost involved in attaining the last element of $\mathbf{T}$, i.e., t_i, is given by $C_{\text{att}}^-(v, b, \mathbf{T})$. Then,

$$C_{\text{att}}^-(v, b, \mathbf{T}) = C_{\text{att}}(v, b, \mathbf{T}) - b_h^i,$$

in which $b_h^i = \min_{b_l \in \mathcal{B}} C_{\text{base}}(t_i, b_l)$ as discussed in Definition 2.

Let $\mathbf{T}'$ be a list containing $\mathbf{T}$ to which a target t_k has been added, i.e., $\mathbf{T}' = [t_1, t_2, ..., t_i, t_k]$. The cost to move from t_i to t_k is represented by $C_{\text{att\_ptp}}(t_i, t_k)$. Then,

$$C_{\text{att}}(v, b, \mathbf{T}') = C_{\text{att}}^-(v, b, \mathbf{T}) + C_{\text{att\_ptp}}(t_i, t_k) + b_p^k$$

in which $b_p^k = \min_{b_l \in \mathcal{B}} C_{\text{base}}(t_k, b_l)$.

Then, $C_{\text{att}}(.)$ is monotonous non-decreasing iff

$$\begin{aligned} C_{\text{att}}(v, b, \mathbf{T}') \geq C_{\text{att}}(v, b, \mathbf{T}) \quad &\Leftrightarrow \\ C_{\text{att\_ptp}}(t_i, t_k) + b_p^k \geq b_h^i \quad &\Leftrightarrow \\ C_{\text{att\_ptp}}(t_i, t_k) + b_p^k \geq b_p^i \quad & \qquad (1) \end{aligned}$$

Then, by Definition 2, equation (1) holds. Hence, C_{att} is monotonous non-decreasing. (Note, if the cost to move from t_i, via t_k, to a base would be smaller than b_p^i, this would be in contradiction with Definition 2 item 2, since b_p^i is the minimal cost to base b_p from target t_i.)

Let $C_{\mathbf{T}} = \sum_{t_l \in \mathbf{T}} C_{\text{action}}(v, t_l)$ be the cost involved in performing the actions on each target in the target list $\mathbf{T}$ by vehicle v. Then, $C_{\mathbf{T}'} = C_{\mathbf{T}} + C_{\text{action}}(v, t_k)$. It should be clear that $C_{\mathbf{T}'} \geq C_{\mathbf{T}}$, since $C_{\text{action}} \geq 0$. Hence, C_{action} is monotonous non-decreasing.

Since C_{att} and C_{action} are both monotonous non-decreasing, cost function $C(.)$ is monotonous non-decreasing. □

Corollary 1: Let L and L' be the lists of $X_i's$ corresponding to a plan P and P', respectively. Plan P' subsumes plan P, $P \sqsubseteq P'$., if L is a sublist of L'. Then, $\forall P \sqsubseteq P' : C(P) \leq C(P')$

Proof: Trivial □.

In the next section, we exploit this result in solving the before-mentioned problem.

4 Approach

This section is devoted to the generation and maintenance of plans. In Section 4.1, we introduce a two step approach to select complete plans, in which vehicles are deployed to targets, with acceptable cost. It is up to the expert to decide whether the cost of a plan is acceptable or not. For example, an expert may define a threshold value for the cost of a plan, i.e., all plans that have a lower cost than the threshold value are acceptable. Since the information with regard to a target may be uncertain and/or incomplete, a plan may become invalid or to expensive whenever new information becomes available. In Section 4.2, we address this issue.

4.1 Plan selection

As noted already, we propose a two step approach to select plans that deploy vehicles to targets. In the first step, we generate lists of targets attainable by each vehicle. In the second step, we select for each vehicle at most one list such that the result will be a complete plan with acceptable cost. Theoretically, both steps have an exponential complexity. Therefore, we introduce a number of rules to control the complexity. Let us elaborate the steps in more detail.

Step 1: To generate lists of attainable targets by a vehicle, we rely on the following observations. First, not all vehicles will be suited to each target. Depending on the target characteristics, one can decide whether a vehicle is a candidate to be deployed to a target or not. This information is provided by an expert or by a database. Once this information is available, we generate lists of targets that is longer than 1, taking into account the following principles:

P1: If a list $\mathbf{T}$ is *not* attainable by a vehicle v, then each $\mathbf{T}'$ that contains $\mathbf{T}$ is also not attainable by vehicle v.

P2: If a list $\mathbf{T}$ is attainable by a vehicle v, then each list $\mathbf{T}^-$ that is a sublist of $\mathbf{T}$ is attainable by vehicle v.

The rationale behind these principles follows directly from Definition 2.

P3: If a vehicle v is not able to perform the actions on a list $\mathbf{T}$, i.e., $\sum_{t \in \mathbf{T}} C_{\text{action}}(v, \mathbf{T}) = \infty$, then v is *not* able to perform the actions on any list $\mathbf{T}'$ that contains $\mathbf{T}$.

P4: If a vehicle v is able to perform the actions on a list $\mathbf{T}$, then v is able to perform the actions on any list list $\mathbf{T}^-$ that is a sublist of $\mathbf{T}$.

The rationale behind these principles follows from the fact that C_{action} is monotonous nondecreasing.

Let us illustrate how these principles are used to identify attainable lists for a vehicle that is able to perform the actions on the lists. In the following, such a list is called a *feasible* list for a vehicle.

Example 2 Consider a vehicle v that is deployable to the targets t_1, t_2, and t_3. In Figure 1, it is illustrated how all lists may be generated with regard to the three targets[1]. In general, the number of lists for n targets is $\sum_{k=1}^{n} k! \begin{pmatrix} n \\ k \end{pmatrix}$. Suppose that $[t_1], [t_2]$, and $[t_3]$ are feasible lists for vehicle v. Then, we choose two other lists $\mathbf{T}_1$ and $\mathbf{T}_2$ from Figure 1, such that neither $\mathbf{T}_1$ is a sublist of $\mathbf{T}_2$, nor $\mathbf{T}_2$ is a sublist of $\mathbf{T}_1$, and decide whether the lists are feasible for v or not. Let assume that the chosen lists are $[t_1, t_2]$ and $[t_3, t_2, t_1]$. Suppose that $[t_1, t_2]$ appears not to be attainable by vehicle v, (and, therefore, not feasible for v), and $[t_3, t_2, t_1]$ appears to be feasible for v. This means that all lists that have $[t_1, t_2]$ as sublist, i.e., $[t_1, t_2, t_3]$ and $[t_3, t_1, t_2]$, are not attainable according to principle P1, and, therefore, not feasible for v.

According to principle P2, all sublists of $[t_3, t_2, t_1]$, i.e., $[t_3, t_2]$ and $[t_2, t_1]$ are attainable lists and according to P4 vehicle v is also able to perform the defined actions on these lists. Hence, the lists $[t_3, t_2]$ and $[t_2, t_1]$ are feasible for v. The lists that are candidate for further consideration are surrounded by a box in Figure 2.

Let us investigate the lists $[t_1, t_3]$ and $[t_2, t_3, t_1]$. Suppose that it appears that v is not able to perform the defined actions on $[t_1, t_3]$ and that $[t_2, t_3, t_1]$ is feasible. Then, according to P3, v will not be able to perform the defined actions on the lists $[t_1, t_3, t_2]$ and $[t_2, t_1, t_3]$. Hence, these lists are not feasible for v. Since $[t_2, t_3, t_1]$ is feasible, the lists $[t_2, t_3]$ and $[t_3, t_1]$ are feasible as well according to P2 and P4.

So, the feasible lists for v are: $[t_1], [t_2], [t_3], [t_2, t_1], [t_2, t_3], [t_3, t_1], [t_3, t_2], [t_2, t_3, t_1]$, and $[t_3, t_2, t_1]$. $\square$

Once all feasible lists of targets have been generated for each vehicle, we check if all targets are handled by the available vehicle. If this is the case, we execute step 2 (see below). Otherwise, we select for each target that can not be handled by a single vehicle, the sets of vehicle that are capable to handle this target. Again, we will use variants of principle P1 and P2 to identify these sets. Namely, the following holds: if a set of vehicles V is able to handle a target t, then each superset V' of V is able to handle target t, and if a set of vehicles V is not able to handle a target t, no subset V^- of V is able to handle target t. However, we will consider in our approach for each target the sets with the minimal number of vehicles. In Table 1, an example of the output of step 1 is given (for further clarification see step 2).

We note that the variant of P2 helps in concluding that no complete plan exists. Suppose that the set $\mathcal{V} = \{v_1, v_2, v_3\}$ of vehicles is available, and that the set V is not capable in handling one of the targets t. Then, target t can not be handled by any set of vehicles. Hence, there exists no complete plan.

[1] The notation in the figure slightly differs from notation in the text. t*i*, in which i is a number, in the figure corresponds to t_i in the text.

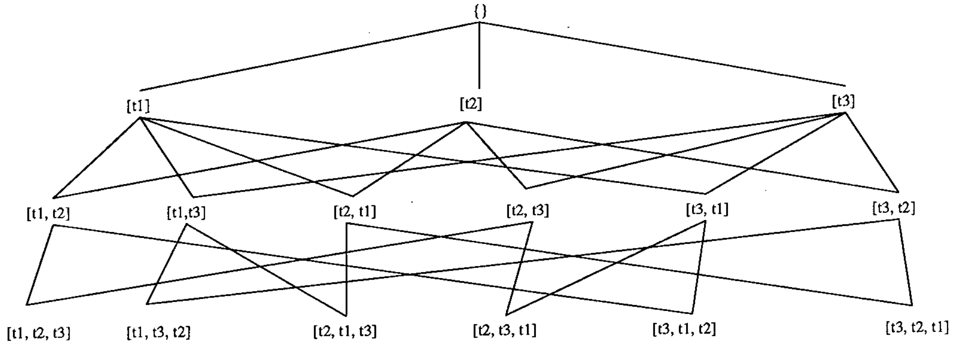


Figure 1: Generation of target lists

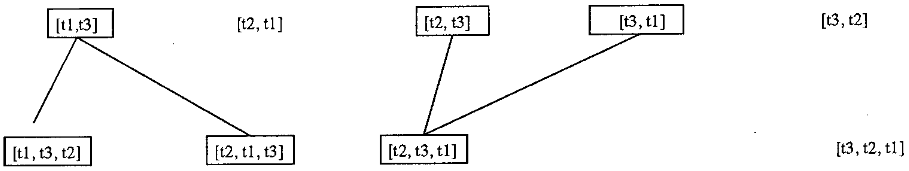


Figure 2: Reduced set of candidate target lists

v_1 : $[t_1], [t_2], [t_3], [t_2, t_1], [t_2, t_3], [t_3, t_1], [t_3, t_2], [t_2, t_3, t_1], [t_3, t_2, t_1]$	$\rightarrow X_1, X_2, X_3, ..., X_9$
v_2 : $[t_1], [t_2], [t_1, t_2], [t_2, t_1]$	$\rightarrow X_{10}, X_{11}, X_{12}, X_{13}$
v_3 $[t_1], [t_3], [t_3, t_1]$	$\rightarrow X_{14}, X_{15}, X_{16}$
v_4 : $[t_1], [t_2], [t_3], [t_5], [t_2, t_5], [t_3, t_5]$	$\rightarrow X_{17}, X_{18}, X_{19}, ..., X_{22}$
t_4 : $\{v_1, v_3\}, \{v_2, v_3\}$	$\rightarrow X_{23}, X_{24}$

Table 1: Feasible target lists for vehicles v_1 to v_4 as output of step1

Step 2: In step 2, a tree is generated on the basis of the results of step 1. Each node in the tree represents an X_i, and paths in a tree represents a plan. Since Proposition 1 holds, the paths corresponding to complete plans is generated according to the branch and bound technique. Let us describe the actions involved in generating the tree.

Consider a set $\{X_1, X_2, X_3, ..., X_n\}$, in which $X_i = \{(V_i, T_i) | i = 1, 2, .., \}$ and T_i is a feasible list for A_i.

1. A random plan that satisfies to Definition 1 is generated, and its cost is computed. This cost will be used to bound parts of the tree and will be referred as *b_cost.*
2. The root of the tree is $X_0 = (\{\}, [])$, and $X_1, X_2, X_3, ..., X_n$ represents the nodes on level 1 of the tree. Then, for each node X_i, we generate a subtree according to action 3.
3. A node X_k is added to a node X_j of a plan $P = [X_0, X_1, ..., X_j]$ as long as the new plan is not in conflict with Definition 1. Nodes are added according to the following rule:

   $$P \cup X_k = \begin{cases} [X_0, X_1, ..., X_j, X_k] & \text{if } X_j \text{ should be} \\ & \text{followed by } X_k \\ [X_0, X_1, ..., \{X_j \cup X_k\}] & \text{otherwise} \end{cases}$$

   If the addition of X_k to P violates Definition 1, we do not expand plan P further, and this part of the tree is bound. Otherwise, the cost of the plan is computed and checked whether the plan is complete or not. If the plan is complete and the cost c is less than *b_cost*, then *b_cost* takes the value of c and the tree is bound from X_k . If the cost c of the plan is greater or equal than *b_cost*, then the tree is bound from X_k as well. Otherwise, action 3 is repeated[2].
4. The procedure terminates if the whole tree has been searched for, or a user defined criterion has been met, e.g. a maximum amount of time.

In the following, we illustrate the steps of the procedure by means of an example.

Example 3 Consider a target set $\{t_1, t_2, t_3, t_4, t_5\}$ and a set of vehicles $\{v_1, v_2, v_3, v_4\}$. In Table 1, the lists of targets that are feasible by a set of vehicles are given as a result of step 1. Note that X_1 corresponds to $(\{v_1\}, [t_1])$, X_9 corresponds to $(\{v_1\}, [t_3, t_2, t_1])$, and so on. X_{23} and X_{24} correspond to $(\{v_1, v_3\}, [t_4])$ and $(\{v_3, v_4\}, [t_4])$, respectively. Suppose that the cost yielded by a plan $P = [X_{12}, \{X_{22} \cup X_{23}\}] = [\{(\{v_2\}, [t_1, t_2])\}, \{(\{v_4\}, [t_3, t_5]), (\{v_1, v_3\}, [t_4])\}]$ is 30, the initial value of *b_cost.*

In Figure 3, a part of the tree has been generated and the cost associated with each plan is surrounded by a box. For example, the cost of plan $[X_{24}]$ is 15. Adding X_9 to this plan, which results into $[X_{24}, X_9]$, increases the cost to 30. Since this cost is equal to the cost of $P = [X_{12}, \{X_{22} \cup X_{23}\}]$, this part of the tree can be bound as a result of Proposition 1. Expanding plan $[X_{24}]$ with X_8 increases the cost to 20, which justifies the further exploration from X_8 of the tree. The result is given in Figure 3. As we can see the plans $[X_{24}, X_8, X_{17}]$, $[X_{24}, X_8, X_{18}]$, and $[X_{24}, X_8, X_{19}]$ are incomplete. Since the cost of plan $[X_{24}, X_8, X_{20}]$ is 25, this is the cheapest plan until now, and *b_cost* gets the value 25.

We note that the expansion of X_{24} with the pairs X_{10} to X_{16} are omitted, since it is in conflict with Definition 1. □

In the next section, we discuss the impact on a plan when previously unknown information with regard to a target becomes available. Furthermore, we discuss how to revise a plan such that it will be in agreement with the new situation.

4.2 Plan updating

As stated in the foregoing, information with regard to targets is the product of continuous intelligence gathering. It may happen that previously unknown information becomes available or that some information becomes invalid in the course of time. For example, a previously unknown target has been discovered or the location of a target that has been assumed appears not to be correct. This may has as consequence that our selected plan may become incomplete or that the actual cost of the plan is more expensive than computed. Therefore, we revise a selected plan in accordance with new relevant information.

For the sake of plan revisions, we store the results of steps 1 and 2, i.e., the attainable list of

[2] A user or a knowledge base may decide whether an action set X_j should precede or follow a set X_k.

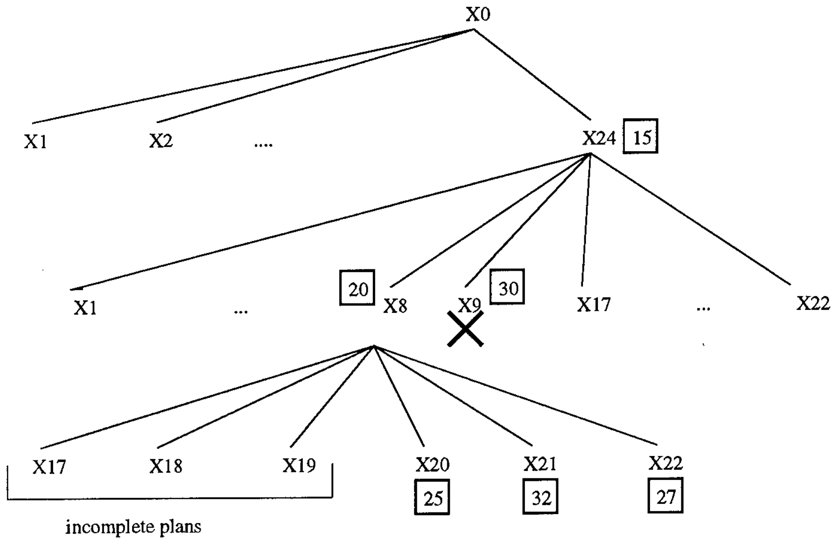


Figure 3: Plan generation

targets by each vehicle and the tree generated in step 2. For the time being, we assume that there is enough storage space to store the tree. Furthermore, we assume that the release of previously unknown information may lead to the following operations in our model.

- deletion of a target t
- insertion of a target t
- update of recorded information with regard to a target t

Let us study the effect of these operations on the results of step 1. If a target t is deleted, then each attainable list in which t appears should be deleted from the results of step 1. If a target t is inserted, then we generate, for each vehicle, all attainable lists that includes t. Update of information with regard to a vehicle can be considered as a deletion of a target followed by an insertion of the same target with the appropriate information.

Let X be the set of X_i's that is the result of step 1. Then, application of a number of abovementioned operations induce to the following situations:

1. X remains the same
2. a non-empty set of X_i's has been removed from X
3. a non-empty set of X_i's has been added to X
4. a combination of 2 and 3 occurs

If X remains the same or a non empty set of X_i's has been removed, then the selected plan is still complete. In the other cases, the plan may become incomplete. However, in all cases it may occur that the cost of the selected plan is changed. In the following, we discuss how the tree generated in step 2 may updated for the different situation.

In the first situation, we recompute the cost at all relevant nodes, i.e., nodes in which target t is involved. This may have as consequence that at nodes where we had decided to bound the tree should be expanded. Suppose that the location of t_2 in Example 3 has been changed and due to this change the cost of plan $[X_{24}, X_8, X_{20}]$ becomes 28 and the cost of plan $[X_{24}, X_9]$ becomes 25. Then, if $b\_cost = 28$, we have no justification to bound the tree at X_9.

To update the tree due to the second situation, we mark all plans, in which at least one of the X_i's appear, as not applicable and the cost of remaining plans are recomputed.

If new X_i's are added in step 1, then the tree is expanded with the new X_i's as illustrated in Figure 4 and the cost at the relevant nodes are recomputed. In Figure 4, the tree of Figure 3 is expanded with X_{25} and X_{26}. We note that if X_i's are added due to insertion of a new target, then the new added nodes are the relevant ones.

Finally, the tree can be updated properly due to situation 4 by first marking the relevant nodes as not applicable, and then adding the new rele-

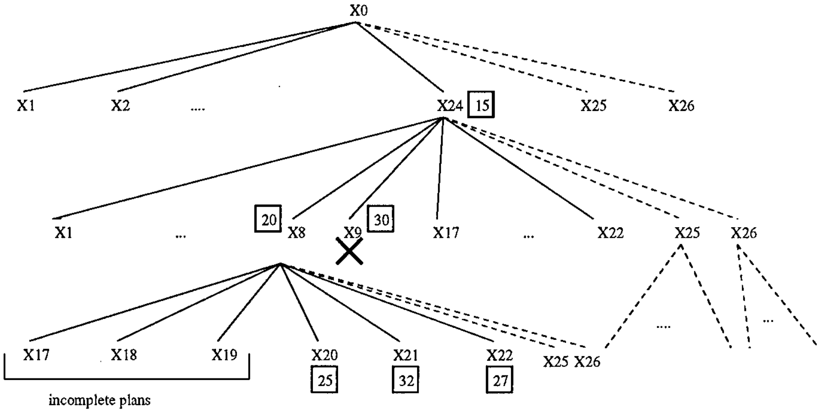


Figure 4: Plan updating due to insertion of pairs X_{25} and X_{26}

vant nodes in the tree.

5 Application

Although our approach is applicable in many domains, we have selected to tackle the so-called resource allocation problem for the naval domain. In this section, we describe how our approach can be applied to solve this problem. Although we are currently implementing our approach for this problem, we are able to provide some preliminary results. These results are based on a preliminary implementation and evaluation. After introducing the resource allocation problem, we will report our preliminary results.

In the naval domain, forces at sea, consisting of different ships, may require support from aircraft to fulfil their mission. Air support is realized by assigning a number of aircraft with weapons to a sea force. We focus on the distribution of ships and aircraft —equipped with weapons— to the different targets, such that each target is damaged to a desirable extent and with acceptable cost. To determine candidate vehicles to a target, an expert applies a number of rules. The antecedent of these rules takes relevant target characteristics, such as size of a target, goal of a target, etc., and relevant information with regard to the environment, such as weather conditions, etc., into account. In [W404], a number of useful rules are reported. These rules can be used to build up the target lists for each vehicle.

In the following, two types of aircraft are distinguished, namely helicopters and fixed-wing aircraft. So, in this application, the set of vehicles is defined as follows: $\mathcal{V} = \{f_1, f_2, ..., f_{n_f}, h_1, h_2, ..., h_{n_h}, s_1, s_2, ..., s_{n_s}\}$, in which f_i is a fixed-wing aircraft, h_i is a helicopter, and s_i is a ship.

The deployment of each vehicle to a target entails some cost, which depends on the type of the vehicle and the weapons carried by this vehicle. In the following, this cost is defined as follows: $F : \mathcal{V} \times \mathcal{W} \rightarrow \Re^+$, in which $\mathcal{W}$ is a set of weapons. A database is available that contains the cost associated with each relevant pair (v, w), in which $v \in \mathcal{V}$ and $w \in \mathcal{W}$.

Let W be a set of pairs (w, q), in which w represents a weapon type and q the number of this type carried by a vehicle v. So, W is defined as $W = \{(w, q) | w \in \mathcal{W} \wedge q = 1, 2, ..\}$.

$$C_{\text{att}}(v, b, [t_1, t_2, ..., t_n]) = \begin{cases} \sum_{z \in W} z.qF(v, z.w) + & \text{if } [t_1, t_2, ..., t_n] \\ d(b, t_1) + \sum_{i=1}^{n-1} d(t_i, t_{i+1}) & \text{is attainable} \\ +d(b_h, t_n) & \text{by } v \\ \infty & \text{otherwise,} \end{cases}$$

in which $d(t_i, t_j)$ is the distance between two targets and $b_h = \min_{b_l \in \mathcal{B}} C_{\text{base}}(t_n, b_l)$.

The damage of each target is measured by a value between 0 and 1, and a target is considered as damaged whenever a threshold value is obtained or exceeded. The damage function D is defined as follows: $D : 2^{\mathcal{V}} \times 2^{\mathcal{W}} \times \mathcal{T} \rightarrow [0, 1]$. The damage value of a triple $(V \subseteq \mathcal{V}, W \subseteq \mathcal{W}, t)$ is derived from a database as well. Plans that contain targets that are not damaged are not complete.

In this application an additional constraint is put upon plans, namely a plan should be completed within a time interval. It should be clear that this constraint may be used in bounding parts of the plan tree, which is discussed in the previous section. If a plan P is not completed in a certain interval, then no plan that subsumes P

can be completed in this time interval. So, all plans containing P can be ignored for investigation.

We note that, for the time-being, the cost to perform actions on a target is not explicitly specified but included in the function C_{att}.

As noted already, a preliminary implementation has been made for the resource allocation problem, called RACAS, which has been described in [5]. RACAS takes as input a number of vehicles and targets, and produces as output plans with decreasing costs. We have offered RACAS various combination of vehicles and targets and have monitored its results. Two conclusions could be drawn from these results. RACAS had no problems in generating lists of attainable targets, i.e., step 1 of our approach. In performing step 1, some of the available domain knowledge has been applied. Second, the first implementation of RACAS ran into memory problems while generating plans (step 2), since we had decided to store the whole tree generated in step 2 in order to facilitate updates.

To solve the memory problem during step 2, strategies are being devised to remove parts of the tree. Two obvious strategies are: 1) remove those parts of the tree that consist of plans whose costs are higher than a threshold cost value, 2) remove those parts of the tree that have not been used for a while or that is not expected to be used frequently. For example, if two action sets in a plan are not desired, then paths in the tree that contains those sets can be removed. We feel that domain knowledge may play a major role in removing large parts of the tree, and to use the available memory efficiently. These issues are currently under investigation.

6 Conclusions & further research

We have presented a novel approach to tackle VMP. We have integrated mathematical results, which have been derived by formalizing VMP, with domain knowledge to solve the problem. Furthermore, we have chosen a more accessible formalism to express plans. This has as advantage that a plan is better understood by users, which in turn makes it easier to add domain knowledge in the planning process. Traditionally, mission planning problems are formulated as: Optimize a function that is subjected to a set of constraints. Often, the user does not have any insight how this problem is solved, which makes it difficult to add useful domain knowledge. Finally, we have proposed a strategy to update plans according to new information. This strategy keeps replanning efforts to a minimum.

Although currently our approach is being implemented, we were able to perform a preliminary evaluation on the basis of an application in the naval domain. Therefore, a preliminary implementation of some vital parts of our approach has been realized. From this preliminary evaluation, we conclude that step 1 of our approach could be performed successfully for this application without extensively using domain knowledge. Storing the whole tree generated during step 2 led to memory problems, simply because this tree is too large. We have proposed several strategies to tackle this problem.

In the first half of 1998, the preliminary implementation will be extended to a tool for resource allocation in the naval domain that will be equipped with domain knowledge and strategies for an efficient use of memory space.

Acknowledgments The author thanks Yves van de Vijver from NLR and Julio Rives from INDRA DTD (Spain) for their useful comments on earlier drafts of this paper. Niels Basjes from NLR is thanked for his implementation efforts.

References

[1] New Advances in Mission Planning and Rehearsal Systems, AGARD Lecture Series 192, NATO, October 1993.

[2] Dockery, J.T., Woodcock, A.E.R., The Military Landscape, Wood head Publishing, Cambridge, England, 1993.

[3] Donker, J.C., Artificial Intelligence Methods and Systems for Planning and Scheduling, Overview of the State of the Art, Technical report TP 97356 L, National Aerospace Lab., Amsterdam, 1997.

[4] Kolitz, S.E.,Computing Techniques in Mission Planning, in [1], pp. 4.1-4.20.

[5] Euclid RTP 6.1 - Grace, Working Paper W420, To be published as NLR Technical Report.

[6] Euclid RTP 6.1 - Grace, Working Paper W404.2, To be published as NLR Technical Report.

High-Mobility Machine Translation for a Battlefield Environment

V.M Holland
C.D. Schlesiger
U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783
USA

1. PREFACE

ARL has developed a small, lightweight document reader and translator called FALCON (Forward Area Language CONverter), now in prototype. The system couples a laptop computer and scanner with software for character recognition and text-to-text translation. These components are protected in a padded, rugged metal case. FALCON was originally intended to help non-linguists in a forward area of the battlefield assess the significance of captured foreign documents. It has also shown itself to be useful in coalition operations and in peacekeeping missions. Begun by the Army Materiel Command's FAST (Field Activity in Science & Technology) Program in response to a requirement from the 18th Airborne Corps, FALCON has been developed through joint efforts of the U.S. Army Research Laboratory (ARL), the Air Force's National Air Intelligence Center (NAIC), and the intelligence community. Now being tested with U.S. troops in Bosnia, the current version of FALCON weighs 28 pounds and can operate on batteries or field power sources.

2. INTRODUCTION

2.1 FALCON as a Relevance Filter

Military personnel placed in front positions of a foreign mission often encounter printed documents in a language they cannot understand. U.S. peace-keeping operations in Haiti, for example, accumulated vast numbers of documents of unknown intelligence significance because linguists trained in French or Creole were in short supply or were assigned to higher priority missions. Eventually, linguistically trained analysts supporting the mission from a distance either became deluged with documents they did not have time to translate or missed key documents that were never culled and sent back. Non-linguists had no way of helping. This situation is typical of military operations, whether combat or non-combat: Not only is intelligence gathering from real-world paper sources slow and cumbersome, but a valuable military resource – literate and informed troops at the front – is underutilized.

FALCON was designed to bring non-linguists into the loop of the analysis process by giving them a portable document analysis support tool. With FALCON they can scan a printed page, recognize individual characters within the scanned image, and produce a rough English translation to evaluate with keyword searches. They can then identify captured documents that match a profile of keywords defined by analysts for the mission. Documents that pass this relevance filter can be transmitted electronically, along with the translation produced, for further processing by linguists. Trials of a 1995 FALCON prototype in Haiti showed good results for French. In 1996 FALCON was selected by the Army Material Command as one of 14 technologies capable of providing solutions in Bosnia. Revised FALCON prototypes were then tested in Bosnia beginning in 1997.

2.2 FALCON as a MT Testbed

As part of ARL's research program in human language technology, FALCON serves as a testbed for emerging concepts in machine translation and OCR technology. This testbed allows us to quickly integrate advances in multilingual text processing and to assess how well they work for real jobs. We can then get feedback from users on interface, speed, and application requirements. Testbed sites are being set up with military intelligence units and the Special Operations Forces, and we have begun to collect baseline data on the effectiveness of large-domain language translators and tailorable

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

keyword searches for the task of judging document relevance with respect to variable topics.

3. DEVELOPING FALCON

3.1 Hardware components

The primary hardware goal is to develop towards a lighter and smaller package, with increasingly faster prototyping for new design concepts. Issues include balancing the competing goals of durability and portability. For portability requirements, the FALCON system is packaged with the capability to be powered from a variety of sources: AC power in US and European standards of 110 VAC and 220 VAC respectively, 24 VDC external supply for connection to external vehicle batteries, and internal power from standard SINCGARS radio type batteries. Engineers were required to design a power supply system to handle all of these conditions and to balance that requirement with size issues. Additionally, it was required that the system be able to connect to several communication sources. This called for wiring connection ports for modem, network (MSE), and radio (SINCGARS) links. Engineers were faced with the problem of including the above components with a scanner suitable to a wide variety of document types that could be found in the field and with a computer capable of running the machine translation and other software all in a portable, compact container. The components were mounted in a carrying case to keep them stable and protected when traveling, but with a layout that allows a user to simply open the case and power up the system to begin its use.

An early FALCON prototype weighed 45 pounds and was developed in 7 weeks. The prototype being tested in Bosnia weighs 28 pounds and was repackaged in 3 weeks, reflecting improvements in survivability, durability, and usability. The unit contains tools to support some of its own maintenance.

3.2 Software Components

Documents are processed by FALCON in 3 phases, each with a separate software package: (a) *scanning* generates a bitmap image of pages fed into the scanner, (b) *optical character recognition* (OCR) recognizes characters in the bitmap image and generates text files of the recognized characters, (c) *translation* converts source text to English text and supports keyword searches.

The aim has been to insert off-the-shelf packages for scanning and OCR and to integrate commercial-grade translation software designed for PC delivery. Packages meeting these requirements have been selected and refined in collaboration with the U.S. Air Force and intelligence community. The translation software, sponsored by NAIC, is developed by SYSTRAN Inc. Based on a "direct translation" method, this software is augmented for syntactic and morphological analysis and seeks to attain the syntactic level of analysis now found in transfer systems. The translation into English approximates the meaning of the original text but requires much post-editing for fluency. This system illustrates the tradeoff we have made between power, on the one hand, since SYSTRAN methods do not provide full natural language understanding, and portability, on the other, since the software runs fast on a PC. Translation of Serbian and Croatian was developed expressly for use in Bosnia and has a supporting lexicon of 100K words. The direction of future translation is toward more power. As basic advances in machine translation undergo transition to robust performance on affordable platforms, they can be inserted into FALCON. For example, a DARPA-sponsored system that uses statistically based translation in a multi-engine architecture is being integrated into FALCON for 1998 user trials. Similarly, as advances are made in document cleaning and OCR techniques, they can be inserted into FALCON and their effects observed on overall translation quality.

3.3 FALCON Interface

The aim of the FALCON system is to allow non-linguists and non-experts to evaluate foreign language documents in a field environment. To facilitate this goal, ARL has endeavored to make the use of FALCON as simple and quick as possible. Software integration plays a role in this task by reducing the number of software packages that the user has to interact with.

The software integration now underway aims toward a simple, one-button interface for document processing. Steps the user takes in going from scanning to translation were reduced from 19 in an earlier prototype to 5 in the current prototype. At the same time, we have provided flexible paths by which users can correct and edit the results of scanning and OCR to reduce error build-up prior to translation.

To operate FALCON, a user merely needs to indicate which source language the document for processing is written in and which list of keywords are of interest in for searching. Once this is done, the FALCON interface sets all of the relevant program settings for each of the software packages and the user can begin to use the system.

4. EVALUATING FALCON

4.1 User Feedback

After a year of field use in Bosnia, users of FALCON provided ARL with valuable feedback. Non-linguists who used the system found that it was indeed sufficient for document screening. They found that it saved processing time for acquired documents and reduced the workload of linguists. The quality of translation, in their estimation was up to 80% accurate, which was felt to be good enough for efficient screening and evaluation for significance. Unexpected uses for FALCON were also reported. Some linguists were using the system as an aid to their translating work. Some native speakers of Serbian and Croatian were using the system to verify their own translations into English and to provide the English technical terms with which non-native English speakers are often unfamiliar.

With all the good reports about the system, there were a number of limitations noted as well. The users found that the OCR package was not capable of achieving useful accuracy on many documents typical of those found in the field: faxes, mechanically typed, multi-generation copies, or dirty pages. This finding is representative of all commercial OCR: it is geared toward laser-printed pages and not toward low-quality documents. We concluded that basic research is needed to improve the performance of OCR on degraded documents, and we linked with ongoing research in the intelligence community aimed at advancing the state of OCR. Another limitation involved the scanner equipment. For example, users of FALCON encountered stapled pages, identification cards, onion skin paper, and torn pages.

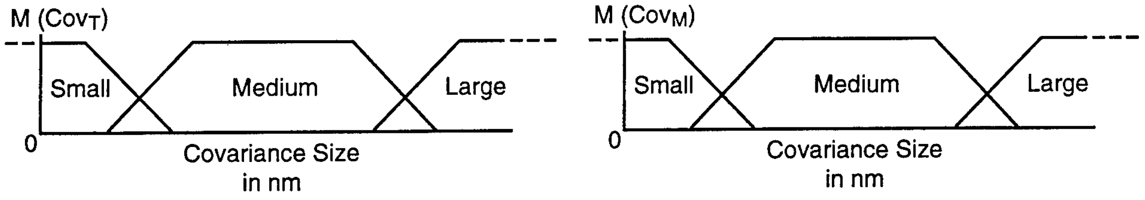
4.2 Metrics for machine translation

Methods for evaluating performance of machine translation systems are evolving. Problems remain to be solved in measuring the accuracy and precision of translations. Addition of scanning and OCR processes further complicates evaluation by introducing more potential sources of error. We are conducting experiments to separate out the error due to these different stages of processing and to find ways to improve accuracy at each stage. FALCON provides a way to benchmark statistical metrics against the judgments of human users on the effectiveness of translations for particular tasks.

5. FUTURE DEVELOPMENT

New prototypes of the FALCON system are in development at ARL. Industry hardware components are available now in much smaller form than when the earlier prototypes were built. Laptop computers are even more lightweight than before and scanners have better resolution in a smaller package. Alternatives to sheet-fed scanners are also becoming more viable as technology improves. ARL is looking towards a new prototype by the end of FY98 with a weight of approaching 20 pounds. During FY99, very small, wearable platforms will be explored to support translation in a battlefield environment.

ARL is working in a number of areas for improved software performance: upgrading to the current machine translation and OCR software; incorporating new approaches to machine translation, which add new language capabilities to FALCON; integrating stronger search methods than keyword searching; evaluating emerging translingual information retrieval tools; and extending the software to handle speech translation.



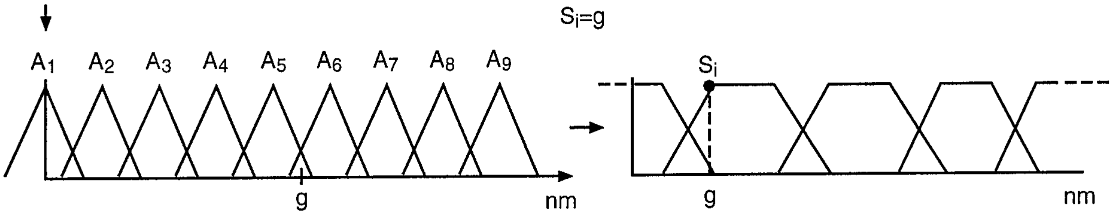
Track Covariance Size Membership Functions

Measurement Covariance Size Membership Functions

Figure 9. Covariance Membership Functions

Track Covariance

Measurement Covariance		S	M	L
	S	A_1	A_2	A_3
	M	A_4	A_5	A_6
	L	A_7	A_8	A_9



Parameter Output Membership Functions

Residual Membership Functions

Figure 10. The Defining Parameters of the Residual Membership Functions are Computed From Covariance Information

Percent of Track Overlap

Percent of Measurement Overlap	Residual Rule: R	None	Very Low	Low	Med	Most	All
	None	X^R_{11}	X^R_{12}	X^R_{13}	X^R_{14}	X^R_{15}	X^R_{16}
	Very Low	X^R_{21}	X^R_{22}	X^R_{23}	X^R_{24}	X^R_{25}	X^R_{26}
	Low	X^R_{31}	X^R_{32}	X^R_{33}	X^R_{34}	X^R_{35}	X^R_{36}
	Med	X^R_{41}	X^R_{42}	X^R_{43}	X^R_{44}	X^R_{45}	X^R_{46}
	Most	X^R_{51}	X^R_{52}	X^R_{53}	X^R_{54}	X^R_{55}	X^R_{56}
	All	X^R_{61}	X^R_{62}	X^R_{63}	X^R_{64}	X^R_{65}	X^R_{66}

Figure 11. Association Inference Engines are Indexed by Residual Size

ACKNOWLEDGMENTS

We thank the US Army MICOM for funding this research and development under contract DAAH01-97-C-R099 under technical supervision of B.H. Ace Roberts.

REFERENCES

1. Bar-Shalom, Y. and Li, X.-R., "Estimation and Tracking: Principles, Techniques, and Software," Norwood, MA, Artech House Inc., 1993.
2. Blackman, S., "Multiple-Target Tracking with Radar Applications," Norwood MA, Artech House, 1986.
3. Garey, M.R. and Johnson, D.S., "Computers and Intractability: A Guide to the Theory of {NP}-completeness," San Francisco, CA, W. H. Freeman and Company, 1979.
4. Hong, L. and Wang, G-J., "Centralized Integration of Multisensor Noisy and Fuzzy Data," preprint, 1992.
5. Mendel, J.M., "Lessons in Digital Estimation Theory," Englewood Cliffs, NJ, Prentice Hall, 1979.
6. Priebe, R. and. Jones, R., "Fuzzy Logic Approach to Multi-Target Tracking in Clutter," SPIE, Acquisition, Tracking, and Pointing V, Vol. 1482, 1991, pp. 265-274.
7. Stubberud, S.C., Lobbia, R.N., and Stubberud, A.R., "Improved State Estimation Using Artificial Neural Networks," in "Proceedings of the Fifth International Conference on Advances in Communication and Control," Chania, Greece, June, 1995.

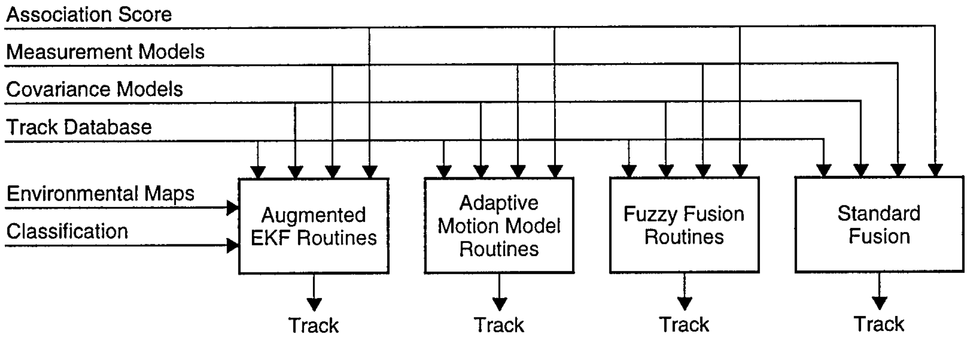


Figure 12. Fusion Implementation Design

8. Stubberud, S.C., Lobbia, R.N., and Owen, M., "An Adaptive Extended Kalman Filter Using Artificial Neural Networks," in "Proceedings of the 34th IEEE Conference on Decision and Control," New Orleans, Louisiana, December, 1995, pp. 1852-1856.

9. Stubberud, S.C. and Kowalski C., "State Vector Quality Assessment," USAF Contract No. F33615-97-C-1063 Final Report, January, 1998.

10. Stubberud, S.C. and Owen, M., "Targeted On-Line Modeling for an Extended Kalman Filter Using Artificial Neural Networks," to be presented at the "1998 World Congress for Computational Intelligence," Anchorage, Alaska, May, 1998.

11. Watkins, F.A., "Fuzzy Engineering," Ph.D. thesis, University of California Irvine, Department of Electrical and Computer Engineering, June, 1994

C4I FOR THE WARRIOR: SUPPORTING OPERATION JOINT ENDEAVOR

Janet Lepanto

Saul Serben

The Charles Stark Draper Laboratory, Inc.

MS 89

555 Technology Square

Cambridge, MA 02139, USA

Introduction

This paper presents an overview of the Bosnia Command and Control Augmentation (BC2A) Initiative, which was directed by the Defense Advanced Research Projects Agency (DARPA)[1], with Draper Laboratory providing the overall system architecture design, implementation, and systems integration. DARPA maintained close coordination with numerous government agencies, the services, and major military commands to ensure that the command and control needs of the warfighter in theater are met. The goal of the BC2A Initiative was to integrate a communications and network infrastructure with information servers and information management software to provide command, control, communications, computers, and information (C4I) capabilities for US and NATO forces in EUCOM and Bosnia. This goal was achieved by: exploiting commercial networking technologies to provide higher bandwidth, better quality information products to the forward deployed warfighter; leveraging the considerable DoD investment in C4I technology to enhance near term operational capabilities; and making the United States' secret intelligence available to warfighters throughout the theater via secure internet connections.

In the BC2A implementation of C4I, the warfighter accesses information and intelligence products via deployable nodes - suites of hardware and software for satellite communications, networking, and information management - that are located at command centers and at down range sites. The design of the nodes reflects the rapid integration of legacy commercial off the shelf technology (COTS) and government off the shelf technology (GOTS) hardware and

[1] In early 1997, the BC2A Initiative began transitioning from DARPA to the Defense Information Systems Agency (DISA).

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

software, to augment the Global Command and Control System (GCCS) leading edge services (LES) with capabilities that initially included video-teleconferencing (VTC), shared whiteboard, and live Predator video. In addition, the nodes were designed to be redeployable, so that they can be moved from one military "hot spot" to another as required.

The BC2A system architecture encompasses communications networks, information management servers, the information and intelligence support centers that provide information products, and the deployed nodes. This architecture was implemented in three phases: (1) integration and testing of deployed node hardware and software; (2) system level testing of CONUS and EUCOM node configurations; and (3) field installation of the deployed nodes in the theater. The challenges of the BC2A Initiative included: the training of military personnel to utilize the deployed nodes in an operational environment; the requirement to maintain multiple levels of security (i.e. US secret and NATO releasable) across the network; and the transition from internet protocol (IP) based command and control to asynchronous transfer mode (ATM), which supports dynamic bandwidth allocation across the BC2A network.

The BC2A Initiative utilized the US operations in Bosnia as a testbed for inserting advanced C4I capabilities into an operational military environment. By exploiting the significant US advantage in C4I technology, the Initiative has enabled substantial increases in communications bandwidth to support collaborative planning and forward staging of large information products such as imagery. In addition, it has provided advanced information transport and management technology by leveraging emerging capabilities from development programs such as DARPA's Information Dissemination Management (IDM) Program.

The BC2A Initiative has rapidly fielded a significant C4I capability into the EUCOM theater of operations. In the process, there have been several equally significant "lessons learned". First, and most fundamental, is that a reliable and robust communications and network infrastructure is essential to any distributed information management and dissemination system. Second, training requirements greatly impact the transition from a "system under development" to a "fielded system". Therefore, the system architecture needs to anticipate the requirement that the operation and maintenance of the system be straightforward and intuitive, to minimize the requirements for training. Third, strict configuration management of hardware and software during the design, development, and deployment phases is required to ensure consistent, reliable system performance. Fourth, ongoing interaction with the system end user is a powerful mechanism for risk reduction in the development of any C4I system. And finally, an operational information management and dissemination system must support the mission of the warfighter end user by providing access to the C4I capability that the warfighter *needs*, as opposed to indiscriminately inundating the warfighter with all of the information and applications that are available.

Capabilities

The BC2A system employs state of the art telecommunications systems to provide near-real-time dissemination of information to: operational military units; command and control elements of the European theater; the services; the Joint Staff; and the National Command Authority.

BC2A builds on the Command, Control, Communications, Computer, and Intelligence for the Warfighter (C4IFTW) smart push/warrior pull concept to rapidly disseminate command and control information simultaneously to multiple users. The BC2A system augments the existing Defense Information Infrastructure (DII) and provides operational commanders with advanced telecommunications and information retrieval capabilities vital to planning and execution for warfighting and for operations other than war. BC2A also provides an opportunity to test concepts for the global broadcast service (GBS) communications architecture in an operational environment.

These capabilities are managed at several echelons within the European theater. The Commander in Chief (CINC) for USEUCOM determines theater information needs, defines requirements for information flow, establishes classification guidelines for information products that originate within the command, recommends information systems for connection to the BC2A system, and ensures the physical security of terminal locations. This CINC also manages the BC2A information flow in coordination with the Joint Information Management Center (JIMC), and operates the information management center in theater. The CINC for US Air Force Europe (USAFE) is responsible for in-theater beta testing of new software and for training system operators and administrators. The local commanders provide administrative and logistic support to personnel operating BC2A system components within their area of operation, and manage communications security (COMSEC) for the very small aperture terminal (VSAT) satellite and joint broadcast services (JBS) systems.

System Architecture

The BC2A architecture exploits commercial networking capabilities to augment the SIPRNET connectivity of current GCCS systems (Figure 1). The BC2A system provides high quality information (e.g. imagery) using the Joint Broadcast System, and links major command centers using the very small aperture terminal (VSAT) satellite system to enable rapid, accurate decision making using collaborative tools. The BC2A architecture integrates information servers in-theater with SECRET servers in the CONUS intelligence community. The architecture also extends CONUS based DISN ATM services into the EUCOM theater and Bosnia.

The early implementation of BC2A command and control functionality was IP-based, but the system quickly transitioned to support multimedia asynchronous transfer mode (ATM). Similarly, the initial IP-based multilevel encryption was also transitioned to ATM using the interim FASTLANE. The current BC2A system has demonstrated ATM via cable, fiber optic, and satellite.

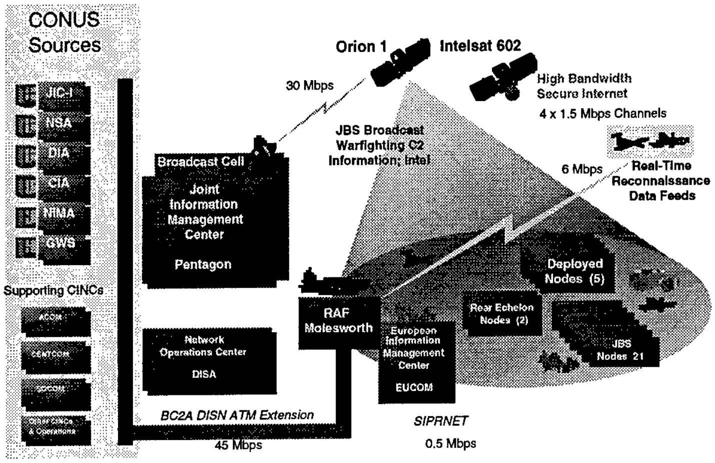


Figure 1. BC2A System Architecture

Communications

The BC2A communications infrastructure comprises the joint broadcast service and the very small aperture terminal network. The JBS provides simultaneous dissemination of data products, in particular high-bandwidth information, to sites in EUCOM; the VSAT network provides point-to-point telecommunications between sites in EUCOM and CONUS.

Joint Broadcast Service (JBS)

The JBS is analogous to the commercial satellite broadcast capability that enables the distribution of information to multiple users over a wide area. The JBS provides the physical communications paths and facilities high-bandwidth delivery (smart push/warrior pull) of information to sites in Europe, including the dissemination of data and video feeds from unmanned aerial vehicles (UAV), and encrypted data or audio files. Cable News Network (CNN™) and Armed Forces Radio and Television Services (AFRTS) broadcasts are also available via the JBS.

The BC2A JBS uplink point is at the JIMC in the Pentagon, and the downlink footprint is located in central Europe. The inter-theater JBS transmissions from CONUS to Europe, and the intra-theater transmissions from USEUCOM headquarters to subordinate elements, are in simplex mode. Intra-theater communications intended for system wide dissemination are routed to CONUS, where they are incorporated into a JBS broadcast. Products can be delivered continuously, periodically, or on demand. Continuous and periodic "push" products are available to any JBS equipped unit in the broadcast footprint. On demand "pull" products are those information products requested by the user. Since the JBS uses receive-only terminals, the user request for information, which may be one-time or recurring, must be submitted through a medium other than JBS.

Very Small Aperture Terminal (VSAT) Network

The VSAT network provides full duplex satellite communications capability between sites equipped with VSAT terminals and receivers. It also provides access to the CONUS DISN-ATM through a virtual theater injection site at the RAF in Molesworth. This network uses a fully interconnected mesh topology to support timely transfer of information between terminals. The VSAT network is used for collaborative planning, including video-teleconferencing and shared white boards; electronic mail; requests from European sites for broadcast data; injecting information collected within the theater (e.g., location of mines, road conditions);and disseminating logistics planning and transportation information.

The VSAT network can support multiple BC2A nodes or sites. The VSAT network currently provides a link between Taszar, Hungary and RAF Molesworth to relay video, voice, and data back to the broadcast management center (BMC) in Washington, DC for rebroadcast over JBS. VSAT terminals are equipped with network encryption system (NES) devices that provide full time encryption of the data paths. These terminals incorporate communications security (COMSEC) devices that allow SECRET material to be tunneled through common US and NATO data paths.

Security

BC2A is currently an accredited command and control system that operates at the SECRET or SECRET-releasable to NATO level, depending on the terminal location and the interconnections with the DISN secure Internet protocol router network (SIPRnet). This is achieved by providing two logically distinct environments. The first operates at the SECRET level and the second at the SECRET-releasable to NATO level. Each environment uses independent information servers and workstations, while sharing a common communications infrastructure. The two environments are cryptologically isolated on the data transmission media.

The two environments interconnect via the C2 guard at the JIMC in the Pentagon and at the theater injection site at Molesworth. Compartmentation is maintained by hardware and software firewalls at both of these facilities. As an added measure, the system at the Pentagon evaluates data at the transport level to ensure that classified data does not flow to an unclassified environment while still allowing data requested by users to flow into the BC2A system. The communications substructure provides a medium to transport bandwidth intensive products (e.g., Predator UAV broadcasts, recorded video) to the JBS uplink point in CONUS. DISN-ATM provides the medium to transport SECRET and SECRET - Releasable to NATO products from CONUS to Europe. Again, compartmentation is maintained cryptographically.

Information Products, Management, and Dissemination

The BC2A system can access a wide variety of information sources. CONUS based sources of information include the Defense Intelligence Agency (DIA), the National Security Agency (NSA), the National Imagery and Mapping Agency (NIMA), the Special Operations Command/Joint Special Operations Command (SOCOM/JSOC), and the Central Command agencies and armed service information centers (CENTCOM). European based sources include the Joint Analysis Center (JAC) in Molesworth; the Combined Air Operations Center (CAOC) in Vicenza, Italy; and the Predator uplink site in Hungary. Unclassified information from the Internet is pulled into the BC2A system via a one way connection.

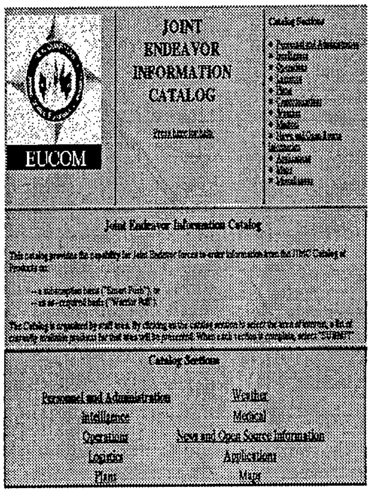
Products currently available include UAV electro-optical and infrared video imagery, digital maps, intelligence imagery, weather graphics, satellite imagery, CNN, AFRTS, and DOD press briefings (Figure 2). Unclassified analog video and audio products, such as live CNN news feeds, or classified digital products are disseminated by the JBS in simplex mode. A portion of the available JBS bandwidth is reserved for data channels. Video imagery from the Predator or Medium Altitude Endurance (MAE) UAVs, or from Navy P-3 aircraft is transported by the VSAT path from the video source to the JBS uplink for injection into the broadcast. Imagery, provided by Air Force U-2 aircraft, can also be transmitted via SIPRNET to the JIMC for re-broadcast by JBS. Currently, Predator imagery is injected at Taszar, Hungary and imagery from the Navy P-3 is injected at Sarajevo, Bosnia-Herzegovina.

Operations
- CJCS
- SITREPs
- Binocular
- CARS

Maps
- DMA
- SPAWAR

Logistics
- DIA

Weather
- AFGWC



JOINT ENDEAVOR INFORMATION CATALOG

EUCOM

Joint Endeavor Information Catalog

Catalog Sections

Personnel and Administration | Weather
Intelligence | Medical
Operations | News and Open Source Information
Logistics | Applications
Plans | Maps

Intelligence
- DIA
- CIA
- NMJIC
- JAC
- FBIS

Administrative
- Early Bird
- Stars & Stripes

IM Notices
- JIMC
- EIMC
- BC2A

Figure 2. BC2A Information Products

The Joint Information Management Center (JIMC), located at the Pentagon, is operated by DISA as part of the Global Management Center (GMC). The JIMC provides all information dissemination services to EUCOM and the warfighter end users. The JIMC also provides a controlled interface between US SECRET and NATO SECRET components of BC2A, between the BC2A system and government agencies, and between the BC2A system and the Internet. Data in any form can be broadcast via the JBS as long as this data can be delivered to the JIMC.

The BC2A system has the capability to disseminate near-real-time, recorded, high-bandwidth, or file transfer protocol (FTP) information products via the JBS and the VSAT satellite network. NTSC standard signals transmitted to the JIMC must be digitally encoded and compressed using either the moving picture experts group (MPEG) or joint photographic experts group (JPEG) compression schemes to minimize bandwidth requirements. Note that when the bandwidth and classification limitations of the system require compression or data encoding of video products, the result may be a loss of resolution. A JBS antenna, an integrated receiver decoder, and a television set are required to receive unencrypted broadcasts. However, a complete JBS suite is required to receive and decode classified video and audio. The VSAT network, including recievers and terminals at user sites, is required to support high-bandwidth, duplex communications among the nodes. The BC2A system transmits data between VSAT terminals for VTC and FTP, and to the JIMC via the JAC for re-broadcast by the JBS.

Deployed Nodes

Each BC2A user site has a deployed node which includes the hardware and software for communications, networking, and information management. At major headquarters the nodes are "relatively permanent" - i.e. they will remain at these locations for the duration of BC2A support - whereas nodes supporting large elements engaged in operations are truly deployable. Each deployed node consists of the communications and computer equipment tailored for the site to receive, transmit, process and display information as required.

There are three configurations for deployed nodes: a command node (Figure 3), a JBS receive only node, and a JBS video receive only node. A command node contains a VSAT terminal, a JBS receiver, and an information server. Each command node is capable of receiving all information disseminated via JBS and supports two-way communications using the VSAT network. A JBS receive only node contains the JBS receive suite and selected terminal equipment. Sites at which a JBS receive only node is deployed can receive the video and data disseminated via JBS. The JBS video receive only nodes contain the equipment needed to receive video. These nodes cannot access the data portion of the JBS broadcast.

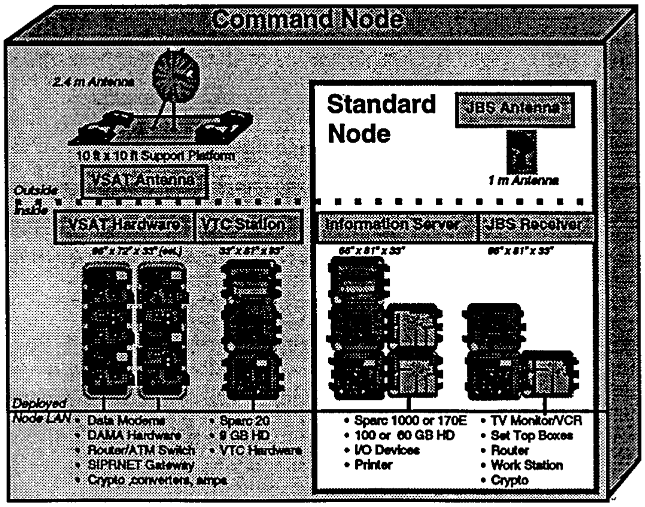


Figure 3. BC2A Command Node

Implementation

The BC2A system was implemented in three phases: (1) component level testing and integration of deployed node hardware and software; (2) system level testing of node configurations in CONUS and EUCOM; and (3) field installation.

Component Level Integration and Testing

Node hardware and software was assembled and tested at the AHPCA. Nodes were customized to include the specific hardware and software necessary to provide the requisite capabilities at each designated site. Software required to support image products archive (IPA), GCCS, JBS, and a web browser was installed and tested on the node workstations. Hardware including disk drives, VTC cameras, and ATM switches was integrated and tested with the workstations. Components of the nodes were packaged in transit cases to ensure that the nodes would withstand the thermal and vibration environment during transport to CONUS test sites and deployment to theater.

CONUS System Level Testing

The CONUS system level tests used a configuration of three nodes to demonstrate network operations, collaboration tools, information management, and communications (Figure 4). The deployed nodes were located at CIA headquarters, USACOM, and the AHPCA, with the JIMC at the Pentagon. The CONUS tests used the actual hardware and software that was subsequently deployed in EUCOM. The system was evaluated to ensure compliance with technical, functional, and operational requirements. First, does the system performance meet design requirements? Second, can the user request, receive and use the information products in a timely fashion? And third, what is the added value of the information products and how can they be utilized?

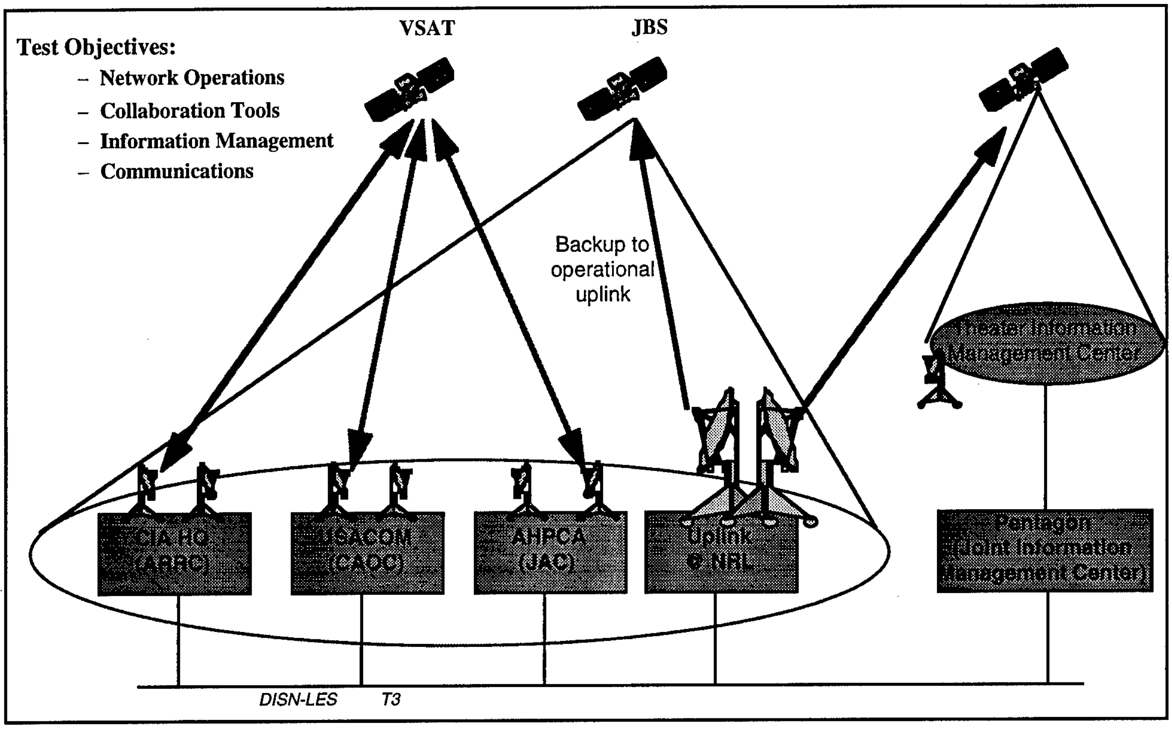


Figure 4. CONUS Test Configuration

The test sessions were attended by representatives from EUCOM, the Joint Chiefs of Staff, and the Secretary of Defense. Demonstrated capabilities included video teleconferencing and dissemination of live video from the Predator UAV. At the conclusion of the successful CONUS tests, the BC2A system was shipped to EUCOM for deployment in an operational environment.

EUCOM Deployment

Site surveys at the designated node locations in EUCOM were completed prior to the deployment of BC2A equipment in EUCOM. The objectives of the surveys were to establish the physical characteristics, configuration of equipment, power requirements, and logistics at each site. A concept of operations (CONOPS) was provided for each of the major component subsystems of the BC2A architecture, as well as for the overall BC2A system. Although training was provided to EUCOM personnel to familiarize them with the system, BC2A personnel - reservist volunteers from all four Services - were available to support and troubleshoot the hardware and software at each user site.

After completion of the CONUS tests, the first three BC2A nodes were shipped to sites in EUCOM, where the system level tests were repeated. These initial sites included the Joint Analysis Center (JAC) in Molesworth, the Combined Air Operations Center (CAOC) in Vicenza, and the Joint Special Operations Task Force (JSOTF) in Brindisi. Following the EUCOM tests, four additional nodes were deployed, beginning with the rear locations (Figure 5). This deployment was consistent with guidance received from the EUCOM CINCs, who stipulated that the deployment and operation of the BC2A system must not under any circumstances interfere with military operations in support of the Bosnia mission.

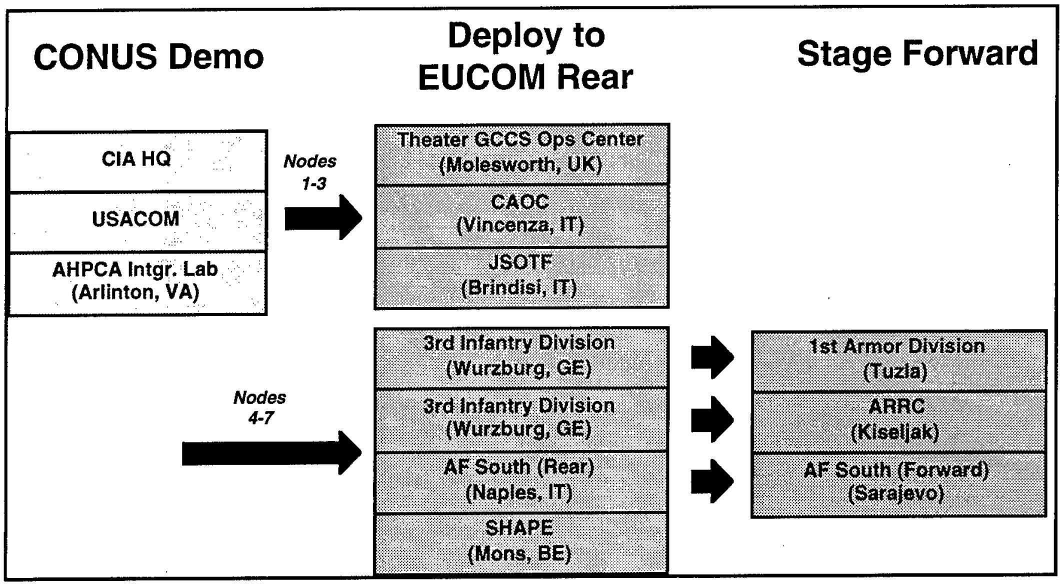

CONUS Demo
Deploy to EUCOM Rear
Stage Forward
CIA HQ
USACOM
AHPCA Intgr. Lab (Arlinton, VA)
Nodes 1-3
Theater GCCS Ops Center (Molesworth, UK)
CAOC (Vincenza, IT)
JSOTF (Brindisi, IT)
Nodes 4-7
3rd Infantry Division (Wurzburg, GE)
3rd Infantry Division (Wurzburg, GE)
AF South (Rear) (Naples, IT)
SHAPE (Mons, BE)
1st Armor Division (Tuzla)
ARRC (Kiseljak)
AF South (Forward) (Sarajevo)

Figure 5. Initial Deployment in EUCOM

BC2A nodes were subsequently deployed and fielded at additional forward sites with the approval and direction of EUCOM (Figure 6).

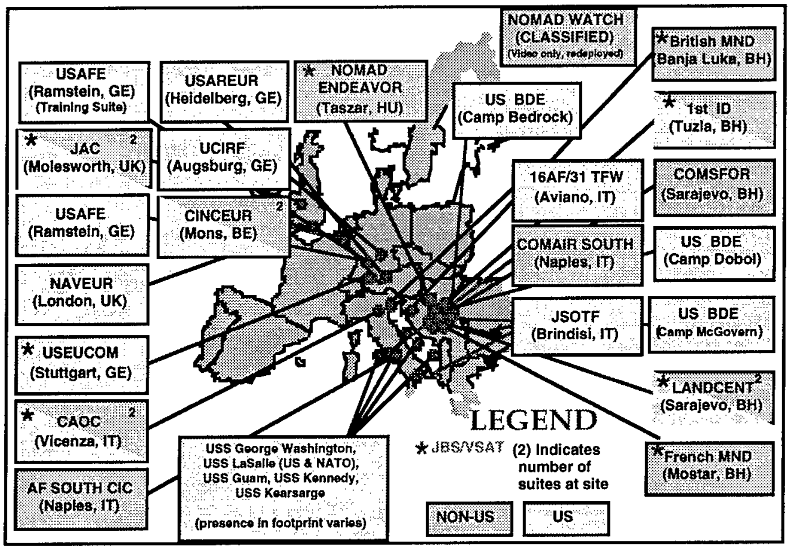

NOMAD WATCH (CLASSIFIED) (Video only, redeployed)
*British MND (Banja Luka, BH)
USAFE (Ramstein, GE) (Training Suite)
USAREUR (Heidelberg, GE)
* NOMAD ENDEAVOR (Taszar, HU)
US BDE (Camp Bedrock)
* 1st ID (Tuzla, BH)
* JAC 2 (Molesworth, UK)
UCIRF (Augsburg, GE)
16AF/31 TFW (Aviano, IT)
COMSFOR (Sarajevo, BH)
USAFE (Ramstein, GE)
CINCEUR 2 (Mons, BE)
COMAIR SOUTH (Naples, IT)
US BDE (Camp Dobol)
NAVEUR (London, UK)
JSOTF (Brindisi, IT)
US BDE (Camp McGovern)
* USEUCOM (Stuttgart, GE)
* LANDCENT 2 (Sarajevo, BH)
* CAOC 2 (Vicenza, IT)
LEGEND
* JBS/VSAT
(2) Indicates number of suites at site
NON-US
US
USS George Washington, USS LaSalle (US & NATO), USS Guam, USS Kennedy, USS Kearsarge (presence in footprint varies)
*French MND (Mostar, BH)
AF SOUTH CIC (Naples, IT)

Figure 6. Deployed Node Sites in EUCOM

Lessons Learned

The rapid development and deployment of the BC2A system was a significant achievement, but this experience also served to underscore the importance of fundamental good engineering practices. Chief among these are configuration management, provision for user training, and interaction with the system end user to ensure that the leave behind system is reliable, usable, and provides the required C4I capabilities.

Configuration Management and System Maintenance

C4I systems employing UNIX or NT are inherently flexible and complex, and when the flexibility of these systems is inappropriately exercised, the result is often the loss of some other system functionality. Systems such as BC2A that rely on the integration of COTS and GOTS applications are particularly susceptible to this problem. Strict configuration management of hardware and software has been imposed to address this problem and to ensure reliability of the BC2A system.

Configuration management and system maintenance activities for BC2A are coordinated by the BC2A Program Office forward (PMO FWD), which is located at Kelley Barracks in Stuttgart. The PMO FWD provides the interface between support personnel at the user sites and the system engineering team in CONUS. The PMO FWD is responsible for overseeing the application of approved BC2A engineering standards for field maintenance, repair, and upgrades.

The focus for BC2A hardware maintenance and repair is the help desk at Kelly Barracks, which logs all reports of hardware malfunctions and coordinates repair actions. Line item replacement units are available at several forward locations. Mobile maintenance teams are dispatched from Sarajevo or Kelley Barracks, as required, when the necessary repairs are beyond the capabilities of the user site operators.

Updates to the BC2A system software are developed, tested, installed, and integrated by the BC2A Information Management team, which is overseen by the BC2A Configuration Control Board (CCB). The site specific configurations of the BC2A software are also controlled by the CCB, although the system administrator at a given site is responsible for performing any modifications.

Training

As BC2A transitioned from a "system under development" to a "fielded operational system", it became clear that formal training of system operators was a recurring requirement due to the routine rotation of military personnel at BC2A user sites. Training for BC2A operators and users is conducted at Ramstein AFB. The BC2A training program combines classroom sessions with on-line interactive training and hands on experience to familiarize personnel with VSAT, JBS, file servers, operations, security, file structures, contingency procedures, and information management and dissemination applications. Training materials are updated regularly to reflect upgrades and revisions to BC2A system hardware and software.

Interaction with the System End User

Regardless of the quality of information and its presentation, if the appropriate information products cannot be moved to the right place at the right time, they are of no benefit to the warfighter. The BC2A system delivers proven capability to the field to meet the warfighter's requirements for C4ISR. The BC2A system enables the warfighter to reliably access relevant, clearly presented information in a timely manner to support situational awareness and the ability to coordinate actions with commanders, subordinates, and peers. The system supports the concept of "smart warrior pull" by providing subscription services for information products. Thus the residual system supports the mission of the warfighter end user by providing access to what the warfighter *needs*, as opposed to inundating the warfighter with what is available.

Acknowledgements

An extensive team of talented and dedicated individuals within industry, government, and the military worked long and hard to ensure the successful design, development, and deployment of the BC2A system. While we purposely refrain from naming specific individuals so as to avoid inadvertently omitting any member of the team, we would like to acknowledge the government program managers COL Ed Mahen (USAF retired) and LTC Al Johnson (USA) for their dedicated efforts on behalf of the BC2A Initiative.

Introducing Machine Intelligence and Autonomy into Satellite Communications Systems

A Krouwel
Strategic Communications and Networks Dept.
Satellite Communications Centre
DERA Defford
Worcestershire WR8 9DU
UK

1. ABSTRACT

The paper takes a high level view of the United Kingdom Military Satellite Communication System (UK MSCS), and starting from first principles determines the key system functions. These functions are then examined for areas where IT support could improve service.

The paper goes on to describe work undertaken at DERA Defford into prototype systems that support each area of operations.

Satellite communications is an area which illustrates many of the problems found in mission systems. Tasks concentrate on the areas of equipment operation, planning, scheduling and responding to unexpected situations.

The paper mentions a number of prototype systems which include,

- Flexible models of communications systems for design, training support, risk reduction, and helping to increase the efficiency of the network and the skills of its operators;
- Automated planning processes for circuits and Radio Frequency (RF) access, based on novel computing techniques, that achieve substantial gains in efficiency, speed and resilience;
- Multiple agent and technique based stress diagnosis and recovery tools designed to work with experienced operators.

For each system some introduction to the problem will be given, and emphasis on how using information technology and machine intelligence can increase the performance of the operator, resulting in better service.

The principle of operation and the state of development of each system is defined, with future research directions and aims presented.

The paper illustrates that IT systems are essential for managing the complexity of future military in a world of increased civil capability, reduced budgets and highly sophisticated opponents.

2. CONCEPT

The UK Military Satellite Communications System (UKMSCS) has been providing satellite communications using the Skynet series of satellites since 1969. Since then the network has changed dramatically, with the latest generation, Skynet 4 stage II, currently being launched.

Each new incarnation of the UK MSCS has brought with it a new level of flexibility in use, new frequency ranges, more flexible antenna configuration options, and so forth.

This increased flexibility has brought with it increasing levels of complexity for the network operators. As in other fields this level of complexity is becoming difficult to manage, and is likely to become more so in the future as we look towards Skynet 5[1] and beyond.

The use of IT systems allows us to simplify and automatically control certain system aspects, reducing this ever increasing complexity load. This gives the system operators more freedom to exercise the additional flexibility available to them to the full, with faster responses to problems and a more robust network.

This paper first presents the functions involved in satellite communications, a brief look at the tasks involved with each function, and how IT systems can be used to support them. It then goes on to describe in more detail systems that demonstrate support to the various functions.

All the systems described in this document are under development for the UK MOD. The aim of these systems is to enable the SATCOM operators to move away from concentrating on low-level tasks, and to enable them to think about more strategic issues.

3. SATELLITE COMMUNICATIONS

The UKMSCS has one mission, which it must fulfil for as much of the time as possible. The mission, stated simply, is to use current and emerging systems to provide service.

[1] The use of the term 'Skynet 5' is used for convenience to signify the next generation system providing UK military communications. It does not signify a preference for any of the currently available development options.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

This primary mission can be split down into several inter-related tasks controlled by the Defence Communications Systems Agency (DCSA). These are,

- Moment to Moment service provision - Supplying uninterrupted, secure, clear communications to the end users from second to second. Includes network patching, production of Control & Monitoring (C&M) data and performance monitoring.
- 'Tactical' planning - Allocating the limited network resources to support the user's communications requirements. Aspects include power planning, network planning and frequency planning.
- Response to Stress (RTS) - Ensuring that services are still available when the network is not performing to its full capability. This can be for a variety of reasons (equipment failure, hostile action etc.) Includes such actions as contingency planning and the use of anti-jamming measures. RTS measures must be ready to operate at any time, and with very little notice.

The primary mission is supported by a secondary task,

- 'Strategic' planning - Longer term planning of the network resources, to ensure that the system can provide its primary function in future years.

In each of these tasks it is possible to offer increased IT support to improve mission performance and service provision at a low cost.

4. MOMENT-TO-MOMENT SERVICE PROVISION

4.1 Summary

Moment-to-moment service provision involves the operation of the network and provision of actual service. The aim of service provision is to ensure that the users communications occur uninterrupted.

4.2 Description

Moment-to-moment operations are largely the domain of the equipment. It is the primary task of the network equipment to actually carry the user's communications. Human involvement at this level involves the manual configuration of certain semi hard-wired equipment settings.

4.3 Solutions

This is the easiest level at which to apply automated measures to increase performance. The simplest level of 'autonomy' is when equipment such as an amplifier fails. Failure is detected automatically, and a redundant unit can be switched into place almost instantly.

The main emphasis at this level is trying to reduce the involvement of comparatively slow human operators, which means reducing the amount of manually configured equipment.

One of the major steps towards achieving this is the planned introduction of separate 'autonomous networks'. Here, the UK MSCS allocates a section of its resources to a system at the tactical planning level. The network control software can then control the moment-to-moment operations. This covers such facilities as Demand Assigned Multiple Access (DAMA) systems, which can reconfigure subsections of the network in seconds to support the current talker in a group, or even establish circuits from one user to another on request.

However, the UK MSCS, like any communications system, contains a certain amount of 'legacy' equipment. These are resources left over from previous system eras that often form such an important part of the network they are not practical, or economic, to dispose of.

This situation presents a potential 'brake' on system development, for example, millisecond level control is not possible if certain items of equipment only report their status every few seconds. In addition, the equipment may not support modern network control protocols, or remote automated configuration.

To tackle some of the problems of legacy equipment a computer based system, the Satellite Access Management System (SAMS) , has been developed to provide computer control and monitoring to older systems.

Essentially SAMS fools the network into seeing the older equipment as more modern than it is. Controls are routed through the SAMS system, translated, and then passed on to the equipment.

An increase in the level of automation and autonomy in the equipment is unavoidable, as it is becoming increasingly difficult to buy equipment without these features. A network containing equipment that can intelligently perform low-level maintenance, report its status to another system, and be controlled from a remote point provides a sound basis for improving higher level functioning.

5. 'TACTICAL' PLANNING

5.1 Summary

Tactical planning involves deploying the available network resources to support the user community's communication needs. Once resources have been allocated in this way they can be controlled by moment-to-moment systems.

5.2 Description

Tactical planning falls into two major categories, representing the two forms of constrained resource. These are the allocation of limited RF-bandwidth to users, Frequency planning, and the allocation of equipment, Network planning.

The traditional form of circuits in the UKMSCS is the 'nailed up' link. This involves the user providing their requirements in an Information Exchange Request (IER) to the UKMSCS operators. This IER contains details of who wants to communicate with whom, where they both are, and what kind of service the user would like.

Network operators then find sufficient resources to support the IER, in the form of equipment, power, and RF bandwidth, taking into account the requirements of all the other current users. Equipment configuration information required by the users is then passed on, and the equipment is set up accordingly.

In producing a resource allocation and route through the network for an IER there are a number of resource constraints.

- There may be insufficient equipment, such as modems or channel cards to support the user;
- There may be insufficient power spare at the UK Satellite Ground Terminal (SGT), or on the satellite itself;
- There may be insufficient RF bandwidth;
- The addition of the new circuit may disrupt existing circuits, forcing them to be replanned.

Due to the diversity of service offered by the UK MSCS planning network configuration is far from straightforward.

Civil telephone networks are based on a homogenous circuit type, with every channel being n*64 kbit/s. Military communications, by contrast, are highly diverse. At the lowest speed we have telegraph services running as low as 50 baud, whereas the same network may have to support videoconferencing or telemedicine at 2Mbit/s and higher. The problem is compounded as each different service uses different amounts of each network resource. The services described above need radically different amounts of bandwidth and power, but despite their differences, both circuits need a single channel card. It is not possible to remove one 2Mbit/s signal and replace it with 32x 64kbit/s baud signals.

This makes communications planning extremely difficult. The effort taken from the moment a large set of IERs are presented, until a plan showing all the equipment layouts is produced can take several man-hours. There is no guarantee that a plan will be optimal or provide good diversity and resilience to equipment failure. Because of the heterogeneous nature of the system, there is not even a rapid way of telling whether the network has enough spare capacity before starting the planning process.

With equipment reconfiguration becoming easier, ways have to be found of increasing the speed and effectiveness of the tactical planning process.

5.3 Solutions

DERA has produced two systems to help in this area, the Network Planning Tool (NPT) and the Frequency Planning Tool (FPT, described in [1]).

5.4 Network Planning Tool (NPT)

The NPT takes a set of IERs and, using a custom constraint-based technique, fits the circuits into the network and allocates resources to them.

The software written in a number of C files, is entirely self-contained when compiled and operates in batch mode. Input files are provided, the software is invoked, and then the results can be read out and used.

The output files produced consist of a file listing each item of equipment and the amount of resource used for each IER and another file ordered by equipment, listing the IERs carried by each item

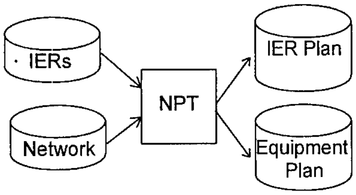


Figure 1 - Software operation

The software itself is extremely rapid. It takes less than a second to produce network plans for a set of over 150 IERs, each of which is made up of several circuits.

The planning process works by considering resources in blocks, as in Figure 2.

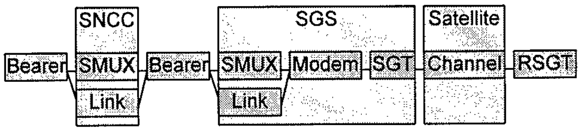


Figure 2 - Circuit routing process

Network planning through a classic UKMSCS network configuration involves the allocation of a certain amount of resources from each of the boxes in Figure 2. Each box also represents a choice, for example, with the UK user on the left and the destination on the right, we have a choice of bearers, and from each bearer we have a choice of which Satellite Network Control Centre (SNCC) to route to. Inside the SNCC there is a choice of link or Switching-multiplexer (SMUX) to pass the data through. From the end of the SNCC there are several bearers to different Satellite Ground Stations (SGS). Each SGS also has a choice of SMUXes or Links then modems and finally a number of Satellite Ground Terminals (SGTs) to send the signal to a satellite. There are also several satellites, each with a number of different channels. Finally there is a remote SGT (RSGT) to connect with the remote user.

The IERs a certain amount of information which does not then have to be planned. For example, the UK user and the remote user are both specified. From the geographical position of the remote user, the correct satellite and channel to use can be determined. This pre-processing reduces a lot of the choices we have to make, and simplifies the problem, as in Figure 7.

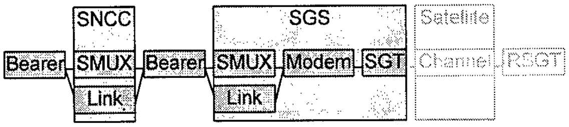


Figure 3 - Circuit routing with IER information

Once the determined areas are eliminated, the process can move on to looking at the choice of other components, taking into account the constraints imposed by the network.

There are three classes of constraints in the system. The first are those based on the connectivity between the components. An example of this is the connectivity between satellites and SGTs. An SGT can only point at one satellite at once, so this constrains the process to only use SGTs pointing to the correct satellite.

Equipment compatibility imposes another class of constraints. For example, the SMUXes can only handle circuits up to a certain datarate. Above this limit direct links must be used instead. The type of modem present at the RSGT also constrains the types of modem that can be allocated at the SGS

The third class of constraints is resource usage, whether there is enough of the required resource left to support the IER. Each of the components is modelled in terms of its limiting resource. In the case of the Bearers and links this is the bandwidth, for the SMUXes it is the bus capacity and the number of channel cards, for the Modems it is simply the number available, and for the SGT and satellite channels it is the available RF power.

Resource usage is reasonably simple to model when planning a single circuit, but difficult to predict when looking at an entire communications set. The amount of a resource remaining in an item of equipment after a circuit has been allocated to it is not necessarily a straightforward relationship. For example, a new circuit is added to an SMUX. The amount of remaining bus capacity is not necessarily the original amount minus the data rate of the circuit. Fragmentation can occur on the SMUX bus, and far more space is denied to other circuits than is used by the recently added one.

Other resource use, such as the transmit power needed at an SGT, depend upon a complex power planning process that takes into account the modem type, the required data rate, the type of remote terminal in use and its position in the satellite antenna footprint. Non-linearities again enter the system as satellite channel amplifier gain tails off when nearing the maximum output (see figure 12)

The planning process begins by looking at all the available items of equipment in the different categories listed in figure 3 once the initial constraints have been applied. From this list the category with the least available resources/choices is selected, i.e. a 'most constrained first' heuristic. For example, if there were two suitable SGTs available, and three different SMUXes, the SGT resource would be allocated first as it is a more constrained choice.

Once a resource is allocated, the constraints are re-checked, and any resources ruled out by the new choice removed from the options. For example, when an SGT is allocated it will be situated at a certain SGS. Resources at other SGSes will then become unavailable.

The process continues selecting and allocating resources until the circuit either has one resource from each box, in which case the circuit is complete, or reaches a point where no resources can be allocated. If the process reaches a stalemate and the circuit is still incomplete it backtracks to the last decision point and tries alternate options. If, after backtracking repeatedly, the process still cannot find a route, the network is considered full and planning stops.

The NPT provides an extremely rapid way of producing network plans. When combined with elements of diversity and capacity planning (as in 7.4) it not only works far faster than existing planning systems, but can produce plans that are more resilient and load-balanced, and hence provide a greater degree of protection to circuits than current planning systems.

Development of more advanced versions of the tactical planning tools is underway, including the addition of a friendlier interface and more routing options. In addition, the next version of both tools will include an interface to response to stress systems (see section 6). This will enable the tools to offer their services in support of response to stress measures to produce rapid re-planning around trouble spots.

Work from the FPT and NPT projects is expected to feed into development of operational tools for the Skynet 5 era network, and strong interest in them has been expressed by UK industry.

6. RESPONSE TO STRESS (RTS)

6.1 Summary

Response to Stress operations occur as a reaction to a reduction in network capacity. The aim of RTS is to restore the capacity as quickly as possible with the minimum of disruption to users.

6.2 Description

Stress, in a satellite communications context, is defined as

Anything which reduces the ability of the system to support communications.

This is a very broad term, encompassing operator error, bad weather, equipment failure, and electronic warfare.

It is important that military satcom systems provide services to the users during the most adverse conditions, as it is at these times that good communications are likely to be most needed.

RTS involves four stages, detection, diagnosis, reconfiguration and recovery. The RTS process concentrates on the last three of these.

The diagnosis of a problem is often far from straightforward. For example, if one item of equipment fails then all the equipment attached to its output may also signal failure. All the equipment attached to those items may then also send out an alarm, and so on. This deluge of alarms can span multiple network sites, and will not necessarily arrive at the operator in the same order that the failures occur.

Once a stress condition is diagnosed RTS operations can move towards reconfiguration, i.e. trying to determine what to do about it. As network design becomes more sophisticated the range of available responses to a given stress situation increases. The strengths and weaknesses of different options are not immediately obvious in a given situation, and in the understandable hurry to restore service it is easy to choose an option that is far from optimal.

Once a reconfiguration option has been selected the recovery process, implementing that option, can begin. A good reconfiguration option minimises the disruption caused at this stage.

6.3 Solutions

DERA is examining ways of assisting the operator by increasing the speed of fault diagnosis, and filtering information.

As well as paper studies to consider various aspects of the problem, many of the ideas are embodied in a piece of demonstration software, The Stress Recovery Assistant. This is designed to reduce the information load on the operator and provide a framework for co-operation.

In addition to direct IT support for RTS the use of system simulators can provide training in a wide range of stress conditions to give operators a broader knowledge of problem areas.

6.4 The Stress Recovery Assistant (SRA)

The UKMSCS Stress Recovery Assistant is designed to provide assistance and advice to operators at all stages of the response to stress process, concentrating particularly on diagnosis and recovery.

The design is a low-risk, modular, distributed, multi-process architecture, based around a series of information 'noticeboards' and specialised 'agents' as in Figure 8.

Each agent is a stand-alone program, that registers itself with a central noticeboard when it starts, using standard communications protocols. This makes the SRA easy to distribute, even in its most basic form. Agents can run on different machines in a network or even across the Internet if desired.

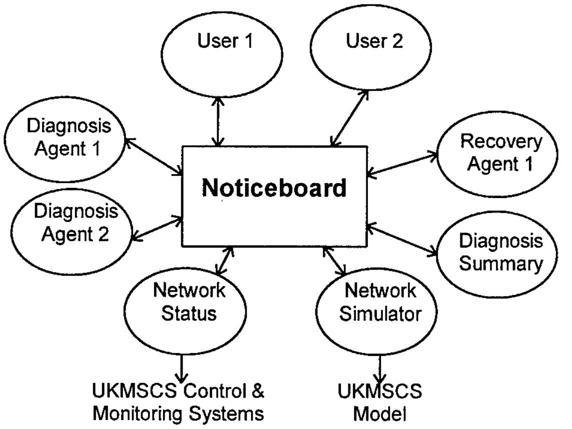


Figure 4 - Example SRA Configuration

Each agent in the SRA provides a number of 'services' to the noticeboard. For example, the Diagnosis agents provide a fault diagnosis service and the Network Simulator provides a service allowing 'what if' scenarios to be executed.

When in operation, agents request services from the noticeboard. When it receives a such a request the noticeboard consults its register of services, and allocates an agent or agents to provide the desired service. Replies are then passed back to the requester via the noticeboard.

As well as this method of working, agents can register an 'interest' in certain parameters and events. For example, the diagnosis agents register an interest in any monitoring data produced by the 'Network Status', as this aids their diagnosis. Whenever the 'Network Status' agent posts monitoring data to the noticeboard a copy will then be passed to each diagnosis agent. This information may cause that agent to produce a diagnosis. This would then be posted to the noticeboard, and passed on to any other interested agents, such as the user.

Agents currently in the system include,

- An agent acting as a proxy for the UK MSCS Control & Monitoring (C&M) system and configuration database. In a real system this would be connected to the actual equipment, and would pass on any alarms or messages generated. In the current system the UK MSCS is simulated by a special near-real time simulator running on the UKMSCS Stress Model (see section 7.5).
- A Network Simulator agent, which provides 'what-if?' simulation runs. Should an agent wish to test a diagnosis to see whether it causes the symptoms observed in the 'real' UK MSCS it can request that this agent run a rapid network simulation with the specified fault. The agent will then receive a simulated stream of monitoring messages from the simulator. This service is provided by a slightly different version of the USM.
- An agent providing a limited user interface. This allows a user to control the SRA, to receive messages, and to examine how it is performing.
- Two agents for producing automated diagnoses of stress conditions, described in more detail below.
- An agent for combining multiple conflicting diagnoses, described below.

The two diagnosis agents included are designed to demonstrate that a mixture of different techniques can be employed to improve the chance of successful operation. The strength of one approach compensates for weaknesses in another.

The first diagnosis agent is an **alarm tracing agent**. This is based on an algorithmic graph tracing process and performs an alarm filtering task required by the multiple messages.

The agent first makes an abstract graph of the current network configuration, and then registers an interest in any monitoring messages from the current network. When a monitoring messages indicating alarms are received the agent begins its diagnosis. For example, at the start of diagnosis, with two alarm messages received, the agent's view of the UK MSCS will look like figure 5.

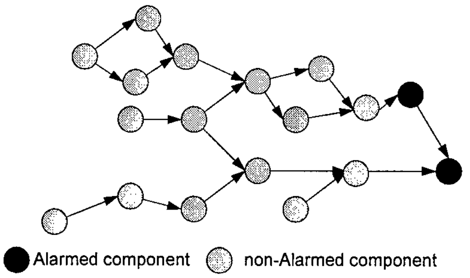


Figure 5 - Example Stress Condition

From this directed graph, the agent can trace all the possible single failure points capable of causing both alarms. In this example the candidates are shown in Figure 6.

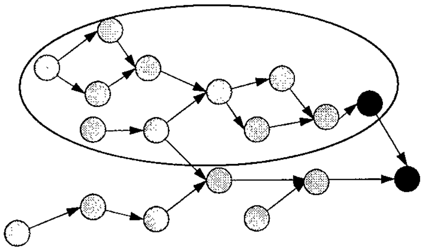


Figure 6 - Possible causes

The agent's diagnosis in the case of Figure 6 will include all the highlighted components. The diagnosis includes a 'confidence' figure which will be higher for those components nearest to the alarms. The agent then posts its diagnosis to the noticeboard, from where it will be sent to interested agents.

Should a new alarm message arrive, the agent can refine its opinion, and re-suggest a diagnosis. If another alarm arrived as shown in Figure 7, for example.

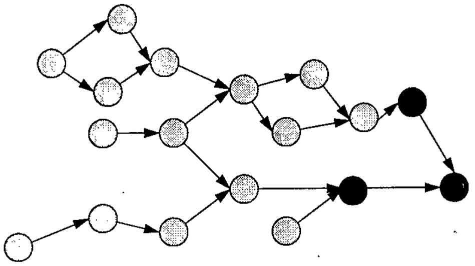


Figure 7 - Additional Component Alarm

This significantly narrows the number of potential single point failures, as shown in figure 8. Note that the new source components were included in the previous diagnosis, but would have been given a lower confidence value as they were quite far from the first possible component.

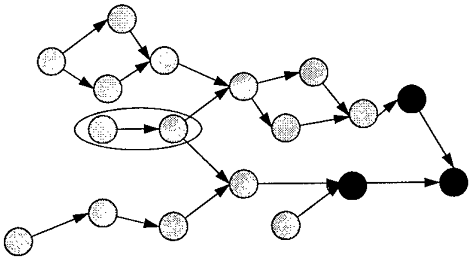


Figure 8 - Amended Diagnosis

The design of the second diagnostic agent uses **case-based reasoning** to pre-simulate the fifty most common network faults.

For each fault the agent requests a simulation of the network with that fault in place from the noticeboard. The noticeboard then passes the request on to the network simulator for execution. The simulator returns the alarm messages that would be received in the event of that fault occurring to the agent.

When a stress occurs in the 'real' system, the agent records the monitoring data it receives from the network status agent. This information is then matched against the simulation symptoms stored in the agent's database. If the agent determines a close match this is returned as the most likely diagnosis. Other alternative diagnoses are returned with confidence based on how close the match is.

The agent also keeps a record of how frequently given faults arise, and can alter its database accordingly, even adding new fault conditions if required.

One of the aims of the SRA is to allow the same problem to be approached from different angles. This is why the two prototype diagnosis agents use different methods to achieve the same ends. Due to this approach the diagnoses produced by the agents may well place different confidence on different failure conditions, as the techniques have different strengths.

To provide an overall view to the operator these alarms are amalgamated by a **diagnostic control agent**. This agent takes all the available diagnoses and attempts to combine them into an overall picture based on the confidence assigned by the diagnosis agents, and the agent's past track record.

If the diagnoses do not broadly agree, the agent can request clarifications from the agents in terms of more detailed diagnoses, or confirmation of whether a certain hypothesised component may be at fault.

The SRA is a highly flexible system currently in its early stages. The major advantage of the architecture is that any program conforming to the interface standard can become an agent, advertise and request services and assist in the recovery process. This allows newly written software, such as the next generation NPT, to be introduced directly into the system, simply by including this interface.

Older and external systems can be interfaced through the use of a proxy agent, which provides an SRA compatible interface. When these systems need replacing, only the proxy agent will need correcting. The other agents will carry on functioning unawares. It is intended that the current network simulator will be upgraded in this manner.

In developmental terms the multi-agent architecture is also useful as not all the agents have to be designed by the same team, or in the same location. The next selection of diagnosis agents to be added to the SRA will come from industrial contractors outside DERA.

The SRA provides a framework for future development, and could theoretically support the user at all levels, including moment-to-moment operations, tactical planning, response to stress and strategic planning.

The integration of all the tools mentioned in this paper will provide a strong footing for the development of future operational systems, demonstrating that such a system could be designed and integrated down to the level of the actual equipment.

7. 'STRATEGIC' PLANNING

7.1 Summary

Strategic planning is a long term role, involving the design of equipment and the provision of network infrastructure.

7.2 Description

Strategic planning, as opposed to tactical planning, involves designing the entirety of the network equipment and operational procedures. Where tactical planning aims to fit IERs into the existing network equipment, strategic planning determines what equipment should be present in the network and how it should be arranged. Strategic planning may involve significant changes to the network resources, for example the addition of new ground terminals or satellites, or the large scale re-arrangement of existing resources, for example the transfer of a large ground terminal from one location to another.

IT support to strategic planning involves improvement in two major areas. One is the evaluation of different network configuration options, to allow objective comparisons of their merits, and the other is in risk reduction to the actual procurement cycle.

7.3 Solutions

DERA has developed systems to help in both areas of strategic planning.

The first system is known as the Future Ground Segment Strategy (FGSS) and was developed to support the evaluation of different potential configurations and equipment arrangements of the UK MSCS.

The second system is a general network simulator which has been designed specifically with procurement support in mind, the Skynet 5 Response To Stress Testbed (or just Testbed.) This system is based on existing work on the UK MSCS Stress Model (USM) which will also be described.

7.4 Future Ground Segment Strategy (FGSS)

This piece of software was developed to meet an urgent need for ways of assessing different possible configuration options in the future UK MSCS. As has happened with a number of systems described here, it was later found to have a much wider application, and spawned the separate NPT (see section 5.4).

The software as specified was to be supplied with a variety of different network architectures, and would be required to test the strengths and weaknesses of each against a typical communications load (IERs) for that period.

To compare the different architectures several different sets of figures would need to be produced. These included a measure of the overall maximum capacity of the network under different redundancy needs, a measure of how the networks could stand up to the loss of one or more major components, and a measure of how much of the optimal capacity could be expected under normal operating conditions.

The program was also expected to compare the performance of the networks using different levels of circuit diversity, to see what improvements could be achieved through giving circuits backup routing. The three available diversity levels were none, partial and full, representing a circuit with no backup, a circuit with a backup routed through a different Satellite Ground Station (SGS), and a circuit with a backup through a different SGS and Satellite Network Control Centre (SNCC) respectively.

To fulfil this requirement three measures were defined

Capacity - How much of the IER set could be supported with that architecture, how much spare capacity remained, and how many backup circuits could be provided.

Survivability - How resistant the network was to failure of major elements, i.e. Network Control Centres and Ground Stations. How much capacity the network could still support with the failure of each individual SGS and SNCC, and with combinations of failures. How much capacity could be diverted to pre-planned backups, and how much could be restored if all the IERs were replanned from scratch.

Availability - How much of the network was likely to be operational for what percentage of time, and what impact would statistical equipment failure have on the network performance.

When designing the software it became apparent that the task of generating metrics was not going to be as easy as a straightforward mathematical analysis of the network. The complexities of the UKMSCS required a hierarchy of planning needs to support the metric generation, as shown in figure 4.

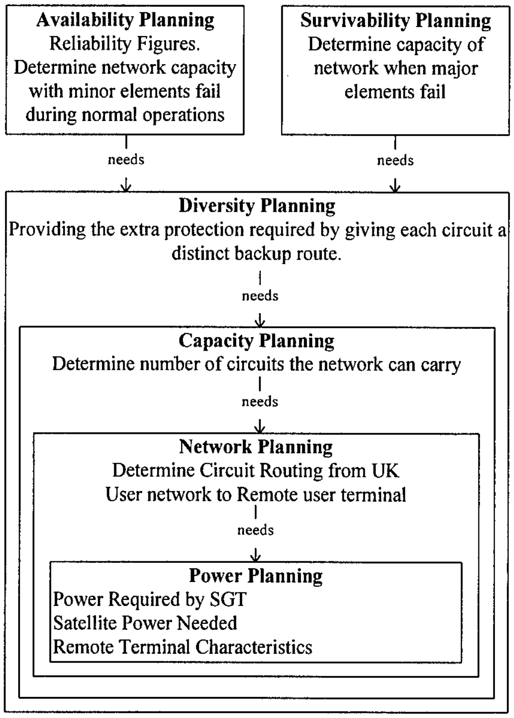

Availability Planning
Reliability Figures.
Determine network capacity with minor elements fail during normal operations
Survivability Planning
Determine capacity of network when major elements fail
needs
needs
Diversity Planning
Providing the extra protection required by giving each circuit a distinct backup route.
needs
Capacity Planning
Determine number of circuits the network can carry
needs
Network Planning
Determine Circuit Routing from UK User network to Remote user terminal
needs
Power Planning
Power Required by SGT
Satellite Power Needed
Remote Terminal Characteristics

Figure 9 - Metric planning dependencies

Availability and Survivability planning (strategic planning) depend on the existence of software that can plan diverse circuits, as these measures need a concept of 'backups'. Diversity in turn requires capacity planning, to achieve load balancing. Due to the complexities of the UKMSCS it is impractical to abstract the capacity planning process, so this also requires software that can plan circuit routes through the network individually. Capacity is then measured by finding the number of circuits we can fit into the network, i.e. network planning, a tactical planning task. This network planning also requires a process that can perform power planning, i.e. the power required at the UK Satellite Ground Terminals (SGTs) to get a clear signal up to the satellite, and down to the remote user's terminal. so that it can calculate resource usage.

Any software designed to provide the requested availability and resilience figures in the UKMSCS needs all the steps outlined above available to it, diversity planning, capacity planning, network planning and power planning.

All these stages are provided by the FGSS software, with the lower levels of power and network planning embodied in the Network Planning Tool. The higher stages were integrated into the lower through the addition of a number of extra constraints to the network planning process.

Once the Network Planning is in place the other elements become relatively straightforward. **Capacity planning** is achieved by counting the number of times the IER set could be planned into the network. Once all the IERs were fitted, the program then tried to plan them again into the remaining resources. The capacity figure is given as a percentage of the IER set fitted.

Diversity planning is implemented by adding additional soft and hard constraints to the Network Planning process, and planning each circuit twice. The new constraints determine that a backup may not be routed through one or more of the same network elements as the primary. For example, in figure 2, partial diversity disallows the choice of the same SGS, and full diversity disallows sharing of the SNCC and SGS.

Additional protection is given through the use of soft 'load balancing' constraints that give guidance when the NPT has to make a choice between two equally valid alternatives. For example, if modems are the most constrained resource, but there are free modems at two SGSes, the process will 'prefer' to choose the SGS that currently has the lesser load.

Availability planning is implemented by an addition to the process. Each item of equipment is given a statistical failure probability based on the chance that it will be available for use at any given time. By running the simulation multiple times, and measuring capacity drops against an ideal network the impact of normal equipment failure can be assessed.

Survivability planning is simply implemented by planning the network, failing a major node, and examining the network plan to see which circuits are affected. For example, once the network is planned one scenario might remove a satellite network control centre. From looking at the NPT outputs it is possible to determine what happens to each IER. Some are unaffected, some circuits are forced to use their pre-planned backup, some are not directly affected but lose their backups, and some are lost altogether. Each of these groups is expressed as a percentage of the IER set for each failure scenario. As an additional measure, all circuits are removed from the network and replanned. This will give a new measure of capacity indicating the drop should that node fail.

The FGSS software was capable of rapidly determining all these metrics. The use of the tactical planning capability to produce network plans gave results that could be fully audited and verified down to the level of individual equipment, giving high confidence.

The further development of the FGSS has concentrated on the separation of the NPT element to form a stand alone system. However, the addition of functionality to produce strategic planning information is fairly straightforward once the network planning element is available, and this will be incorporated in future releases of the NPT.

Any new system would also comply with the SRA standard, providing the facility to assess reconfiguration options in response to stress.

7.5 UKMSCS Stress Model (USM)

The first generation DERA simulator is the UKMSCS Stress Model (USM). This has already been mentioned above supporting aspects of the SRA in the response to stress process, but the USM has established a role in strategic planning that has been taken up with the development of the second generation model, the RTS Testbed. As the Testbed is still at an early stage of development, it is useful to have an understanding of its predecessor.

The USM, was developed in close conjunction with Siemens Plessey Systems[2]. It was designed primarily for Skynet 4 stage I, but has recently been modified for stage II, and is in daily use at DERA Defford.

The USM is a prototype for helping to define the concept and requirements for an operational system. Its primary intended application was in providing 'what-if?' simulations, usually to determine the effect of stress conditions on a particular configuration of the UK MSCS.

The USM presents a logical model of the system at the level of individual items of equipment, such as modems and antennas. It covers both baseband equipment and RF signals, and can model communications from the moment they enter the UKMSCS as digital data until the signals reach the user's terminal in theatre. The model also incorporates a Timing & Frequency (T&F) subsystem, to model synchronisation effects and problems, and a Control & Monitoring (C&M) subsystem to provide automated control of equipment and production of equipment alarms.

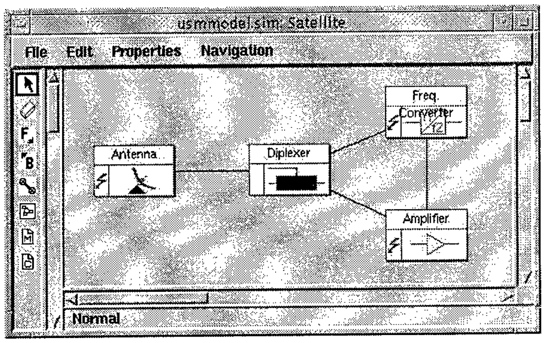


Figure 10 - Example USM Components

Components are modelled in an object-oriented fashion, in terms of their interfaces and their behaviour. Communication between components and the outside world is only allowed

[2] Now British Aerospace Defence Systems Analysis Department

through the tightly defined interfaces, ensuring that each component is self-contained. The behaviour defines the ways in which a component reacts to a signal passed across the interface. For example, the equipment may change its configuration in response to a certain message, or may simply modify the signal and send it out again. As this model was designed to represent stress (i.e. failure) conditions, each component knows how to behave when working correctly, and in a range of failure modes.

The encapsulated design enables early visibility of the system. Very simple components can be entered into the model quickly, and once the system is established, more detail can be implemented for additional realism.

One example of this is the Amplifier component. As we can see in Figure 11, the amplifier has one input and one output.

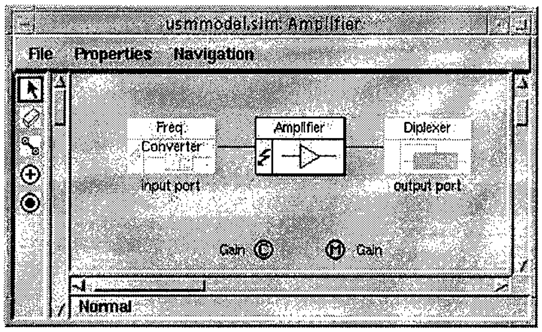


Figure 11 - Amplifier Component

At the simplest level the behaviour of an Amplifier can be defined as increasing the power of the input signal by the amplifier gain. This was the first iteration design.

This, however, is only an approximation of the behaviour of an Amplifier, and as the model progressed it was enhanced. For example, when an Amplifier approaches its maximum output power, it becomes non-linear, as in Figure 12. This means that the increase in signal power is no longer simply the nominal gain. As the amplifier approaches its maximum output power the gain begins to drop off, and the signal produced is not as strong as expected.

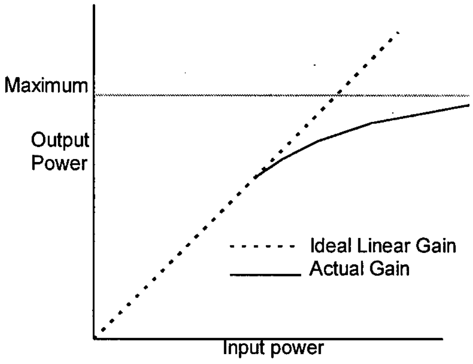


Figure 12 - Linear vs Non-linear Amplifier response

Because each component entirely contains its own behaviour, the Amplifier could be modified to perform this more advanced behaviour without changing any other aspect of the model. The change was therefore incorporated quickly and easily.

The Amplifier was subsequently successfully changed again to create additional non-linear effects, such as Inter-Modulation Products, again building up the component's complexity without having to change any other aspect of the model.

The ability to easily change the USM capabilities and design have proved invaluable in the development of such a prototype. New requirements can be quickly incorporated, and different approaches can be readily tried. The use of an interpreted language (in this case SmallTalk) also allowed modification. without recourse to the usual compile-link-execute cycle, so the simulation could be modified whilst it was running.

7.6 Skynet 5 Response to Stress Testbed

The proposed follow on to the USM is the Skynet 5 Response to Stress Testbed (or Testbed, for short), currently in the requirements definition phase. Building on the experience gained with the USM, this is envisaged as having a far more established role. The Testbed again be will be a system simulator, this time designed for the Skynet 5 era UKMSCS, with full end-to-end connectivity.

The focus of the Testbed is different to its predecessor, being firmly rooted in the Strategic Planning task. The primary role of the Testbed, is as its name suggests, to act as a simulator of Skynet 5 in supporting the development of RTS tools. By having a simulator of the network available before the real network is in service, developers get a chance to test their designs at an early stage, receive feedback, and incorporate improvements early on. By the same measure, the RTS tool customers get early visibility of the RTS systems and can ensure that they are getting what they wanted.

The main problem with developing such a system is that of trying to realistically simulate a system that is not yet defined. The design of the Skynet 5 era network is currently at an early stage and its eventual form is not yet known. In addition, the Testbed must be available before the actual system. This is where the incremental development techniques used in the USM can be used to great advantage.

Although the design of the Skynet 5 era network is not determined we do know that it is likely to contain certain components and capabilities. These can be added to the Testbed in very generic terms early in the design, and as the final form of the actual system takes shape, the components can be more specialised. This will allow the Testbed in some form to be operational long before the system, and so provide well tested RTS support tools from day one of the operations of Skynet 5.

The applications of the Testbed however do not end with initial procurement. It is also being designed with a view to support future strategic planning requirements. In addition it should also be able to fulfil all the other roles currently taken by the USM, including contingency planning and tool support.

One other aspect of the Testbed that is being designed in is with complex, distributed, multi-user simulations in mind. Simulation has been one of the primary services provided by IT to mission systems for many years. A simulator allows users to train, practice, and experience novel situations without risks to themselves or use of expensive hardware. Historically such simulators have been platform based. There are simulators available for tanks, fixed and rotary wing aircraft and missiles, to name a few. Simulation for communications systems has always had a different focus.

Communications system simulators are generally developed for civil use. They concentrate on the execution of a pre-defined statistical model in order to help identify system bottlenecks. They are non-interactive, processor-intensive, and more for use by system designers to analyse the network, than for use in training operators. The FGSS software may be seen as a high level version of this aspect.

The simulation approach we taken here is to try and provide a communications simulation package that is more akin to a tank or aircraft simulator than a network analyser.

This will allow operators to train in the kind of interactive, platform-style simulation used by their colleagues in tanks and aircraft. Testbed facilities will provide both pre-set stress scenarios to test managers and operators, and facilities for 'red-team' operations against human opponents.

The training role does not neatly fall into the functions outlined for satellite communications, but the result of such adversarial simulations may fall into the realm of strategic planning as they allow more structured development of operating policy.

With a simulator in place operator procedures for reacting to certain situations can be refined and improved, and then tested against an opponent, to increase operator, and therefore network, performance.

8. CONCLUSION

The mission of Satellite Communications has been examined from a functional perspective, and broken down into component elements. From these component elements we have seen ways that IT systems can be used to support operations at all levels. Using this functional technique it is straightforward to identify opportunities IT support to mission systems.

The systems described here have been designed to allow users to manage increased complexity whilst increasing the tempo of their operations. Simulators allow the operators experience of novel situations, stress recovery systems filter some of the information deluge and tactical planning tools reduce planning time to effectively instantaneous.

We have also briefly seen examples of different ways of implementing systems that are low-risk, and can be developed in an incremental way. Both the object-based design of the USM and the Agent based design of the SRA provide the benefits of a rough working system early on, with steady improvements in functionality.

There are also examples of how IT systems do not have to be necessarily very advanced to give increased performance. The alarm tracing agent in the SRA, for example, follows a simple algorithmic process, but is surprisingly accurate at fault diagnosis.

The prototype systems developed by DERA Defford have show that IT can be applied to the entire range of system objectives with great benefit. These ideas have attracted a great deal of interest from UK industry, and the future of these systems looks promising.

9. ACKNOWLEDGEMENTS

This work has been funded by MoD(UK). The views represented in this paper are those of the author and do not represent the views of MoD(UK). © British Crown Copyright (1998)/DERA. Published with the permission of the Controller of Her Majesty's Stationary Office.

10. REFERENCES

[1] Using Genetics-based algorithms for Mission Systems Applications, A Krouwel & C Williams, RTO/SCI Symposium on 'The Application of Information Technologies To Mission Systems', Naval Postgraduate School, Monterey, California. April 1998

Although specific references are not supplied for further information if you are interested in any of the work described here, more detailed documentation or demonstrations of any of the systems may be requested via the author, and will be supplied subject to DERA/MoD approval.

The Cognitive Assistant System
and its Contribution to effective Man / Machine Interaction

Frank Ole Flemisch
Reiner Onken
University of Armed Forces Munich
Werner-Heisenberg-Weg 39
D-85577 Neubiberg, Germany
Frank.Flemisch@unibw-muenchen.de
Reiner.Onken@unibw-muenchen.de

Summary

New information technology and highly integrated mission systems offer a high degree of information availibility and powerful machine capabilities.
This can be of great benefit, but might also lead to new problems due to bottlenecks in human information processing, especially in situations with high workload and tough time constraints.
This paper describes a possible solution to cope with this problem: Cognitive Assistant Systems.

CAMA, the Crew Assistant Military Aircraft, is a prototype Cognitive Assistant System for the domain of military transport aircraft. It is under development at the University of Armed Forces Munich in cooperation with DASA, ESG and DLR. Recently it had been tested in a flight simulator with German Airforce pilots and will be tested inflight in early 2000.
Starting with some basic considerations about man / machine interaction, this paper describes the structure of CAMA, its functions, with emphasis on its interface philosophy.

As an outlook caSBAro, an integrated human factors environment for online recording, visualisation, analysis and replay of operator and assistant system behaviour with respect to the underlying situation will be described.

1. Is there in fact a problem?

Information is a key element for success or failure on future battlefields. Continuous advances in information technology and mission systems, especially growing computer capacity and interoperability promise to provide comprehensive tactical situation awareness down to unit level, thereby improving mobility, survivability and sustainability of todays weapon systems.

However increased availability of information may lead to information proliferation and pertinent problems regarding operator information processing under time pressure in a stressfull environment. Are these problems of compulsory nature? Or is there a solution to handle the available overwhelming amount of information? Can the design of the mission management system assist information management without creating additional workload or making existing, robust and up to now successful military procedures ineffective?

Advances in information technology and artificial intelligence also bring along the potential of partially fully automated functions. Is that a problem in itself, or can the socalled function allocation be handled, and can the operator cope with the inevitably rising complexity? When is it of benefit to add new modes and features of automation?

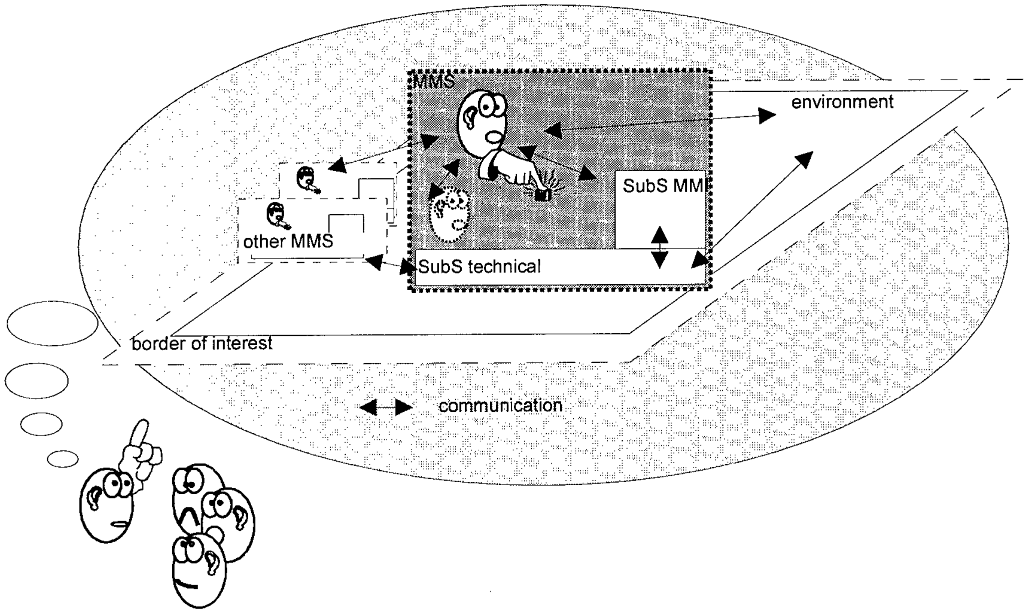
2. A closer look at man/machine interaction

Interaction in a man/machine system is done through communication, this means information transfer across subsystem borders.

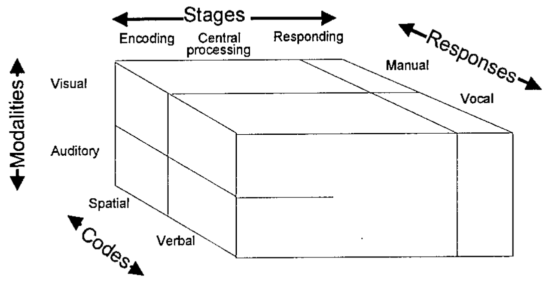
Human computer communication is done by one or more operators on one side and one or more technical subsystems on the other side and is just one part of the overall communication, that also includes the environment (Picture 1).

Information transfer needs communication channels, coding and the information itself.
Communication channels are a connection between an operator's resource [Wickens 92] on one side and a subsystem (e.g. a display) on the other side.
Interaction resources are limited in their number and capacity. Some of these resources can be used in parallel with low interference, others have so much interference that communication on one channel could break down completely if disturbed by another channel (Picture 2).

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.



Picture 1: Communication in a man-machine system



Picture 2. Multiple-Resource Theory [Wickens 92]

Communication events that need identical resources can be **queued** by the operator subject to the cost of delaying certain pieces of communication [Chu & Rouse 1979].
Offering different communication channels (here: modalities) at the same time for one communication event and leaving the decision up to the operator which channel to use can be called **multi-modal** [Taylor et al 1989].

The difference between available communication resources and those needed for a certain task leads to concepts like **workload** [Wickens 1992].

Communication serves certain goals and depends strongly on the situation:

„Human behavior, either cognitive or psychomotor, is too diverse to model unless it is sufficiently constrained by the situation or environment“
[Baron 1984]

The purpose of communication and interaction can be:

- Aligning and confirming the internal models
  - of the world outside the MMS
  - of subsystems within the MMS
  - of the communication itself
- Influencing the future behavior of communication partners (e.g. function allocation, controlling)

Aligning internal models towards reality can be called **situation assessment** and leads to **situation awareness** [Wickens 1996].

The **complexity** of the man/machine system itself must be reflected into mental models. Growing system complexity may lead to higher communication load and to the risk of false internal models. Therefore, high technical complexity must be transformed in a way which can be handled by the operator (virtual machine, [Rauterberg 1996]).

Interaction can be done at different levels of cognition [Rassmussen 83] and can be coded in various manners. The possibility to code and decode information correctly is a resource in itself and needs resources to be performed. A type of coding that is similar to the operator's internal representation and which therefore demands less resources leads to the concepts of „direct perception“ and „**ecological interface**“:

„The design of ecological information systems attempts to exploit the large perceptual and sensory-motor system as encountered in natural environments by furnishing an complex yet transparent information environment. Through a suitable interface, the attempt is made to stimulate a user`s direct perception of information at the means-ends level most appropriate to current needs and, at the same time, support the level of cognitive control at which the user chooses to perform"

[Rasmussen et. al. 1994]

Communication and interaction is also strongly connected to the tasks to be performed. The concept of task **performance** is strongly correlated to the quality of the communication required for that task.

The way how operators control their resources is very much based on expectations delivered by their internal world models. The resulting selection of information transfer and pertinent communication channel leads to the concept of **attention** control.

Communication channels can also attract attention by themselves e.g. through search conspicuity (attention getter), thereby actively influencing the communication process (e.g. concept of visual momentum [Woods 1984]). This can be so strong that the operator's attention is totally consumed by one channel, while all other communication is blocked (**cognitive tunneling**).

Communication is often susceptable to errors. The correct recognition of a communication pattern often depends on expectations and how much disturbance occurs (**cluttering**, signal detection theory [Wickens 1992]).

Interaction can be related to different time scales:

„Pilot's final hurdle is to share resources between short-term behaviour and long term anticipation"
[Wioland & Amalberti 96]

Since resources as well as time are limited, operators take certain risks when deciding what interaction to do next. They control this risk by an internal model (model of ecological safety, ecological safety net [Wioland & Amalberti 96]. Detection and recovery of errors seem to be important to update this fragile model.
The skill to do the right interaction at the right time [attention management skills [FAA 1996]) seems to be of great importance. A lot of accidents especially with aircrafts can at least be partially attributed to a breakdown of these skills.

The concepts of human information processing mentioned above are strongly interrelated to each other and must be considered simultaniously for different perspectives on one complex interaction happening between operator, technical subsystem and the environment.
Most of these concepts were developed for restricted situations. Quantitative assertions regarding complex situations cannot be provided easily.
Nevertheless these concepts help to explain many of the phenomena as being observed in human computer interaction.

What conclusion can be drawn from this discussion for man/machine systems to carry out air missions?

In essence, it tells that more available information and new technical features will only lead to better performance if they account for both the excellencies and the limitations of human information processing. If they don`t, they are exposed to performance drawbacks or even loss of safety.

3. A solution: Cognitive Assistant Systems

Cognitive Assistant Systems (CAS) with autonomous situation assessment, knowledge processing, conflict resolution and information management capabilities respect the excellencies and take into account the limitations of human information processing. They work in partnership with the human operator. They are the answer to many problems mentioned above.

3.1 General philosophy

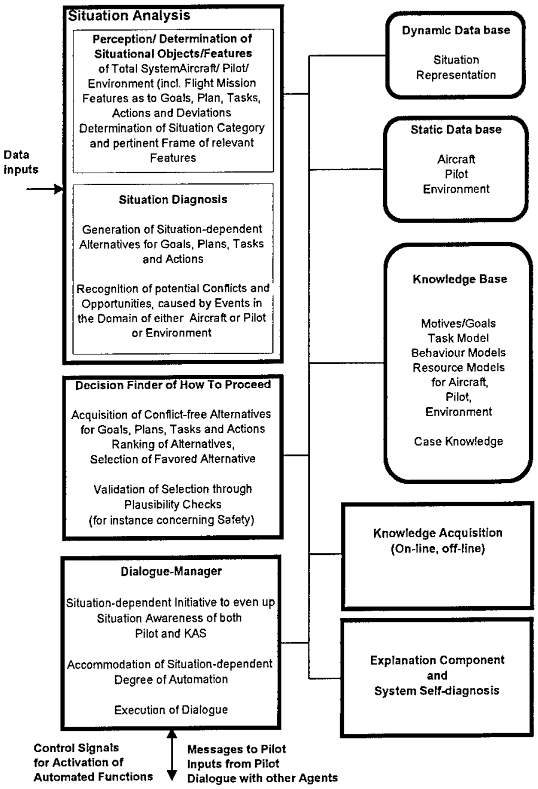
In face of rising problems with automation in glass cockpits of modern aircraft and promising concepts coming along like „human centered automation" [Billings 1991], research at the University of Armed Forces Munich was focussed on Cognitive Assistance Systems.
Considering problems with human information processing limitations mentioned above, two basic requirements for the behavior of such assistant systems can be formulated:

1. Within the presentation of the full picture of the flight situation as necessary for situation awareness it must be ensured that the attention of the cockpit crew is guided towards the objectively most urgent task or subtask of that situation.
2. If (1) is met and there still comes up a situation with overcharge of the cockpit crew, then this situation has to be transferred - by use of technical means - into a situation which can be handled by the crew in a normal manner.

Requirement (1) can only be satifactorily met if situation assessment is performed by the computer in parallel to the human operator. Machine understanding of the situation is the basis to identify relevant tasks and to effectively support the human operator by means of intelligent interfacing.
Requirement (2) can only be achieved if (1) is fullfilled. Automation that is not based on an situation understanding on the machine side will fail to give the right help at the right time.
These considerations lead to a functional layout of an ideal cognitive assistant system as illustrated in picture 3.

3.2 CASSY, a Cognitive Assistant System for IFR aviation

Based on these functional considerations and a first development called ASPIO (Assistant for Single Pilot IFR Operations), a corresponding system for civil IFR operation in a 2-man cockpit CASSY (Cockpit Assistant System) was designed with support by DASA.
CASSY starts with a situation analysis and a flight plan created with application of the situation information (Automatic Flight Planner). Based on a flight plan agreed upon by the pilot, the corresponding behavior of the aircraft / pilot system to be expected is generated (Petri net based Piloting Expert) and compared with the actual behavior (Pilot Intent and Error Recognition).


Situation Analysis
Perception/ Determination of Situational Objects/Features of Total SystemAircraft/ Pilot/ Environment (incl. Flight Mission Features as to Goals, Plan, Tasks, Actions and Deviations
Determination of Situation Category and pertinent Frame of relevant Features
Situation Diagnosis
Generation of Situation-dependent Alternatives for Goals, Plans, Tasks and Actions
Recognition of potential Conflicts and Opportunities, caused by Events in the Domain of either Aircraft or Pilot or Environment
Data inputs
Decision Finder of How To Proceed
Acquisition of Conflict-free Alternatives for Goals, Plans, Tasks and Actions
Ranking of Alternatives,
Selection of Favored Alternative
Validation of Selection through Plausibility Checks (for instance concerning Safety)
Dialogue-Manager
Situation-dependent Initiative to even up Situation Awareness of both Pilot and KAS
Accommodation of Situation-dependent Degree of Automation
Execution of Dialogue
Control Signals for Activation of Automated Functions
Messages to Pilot
Inputs from Pilot
Dialogue with other Agents
Dynamic Data base
Situation Representation
Static Data base
Aircraft
Pilot
Environment
Knowledge Base
Motives/Goals
Task Model
Behaviour Models
Resource Models for Aircraft, Pilot, Environment
Case Knowledge
Knowledge Acquisition (On-line, off-line)
Explanation Component and System Self-diagnosis

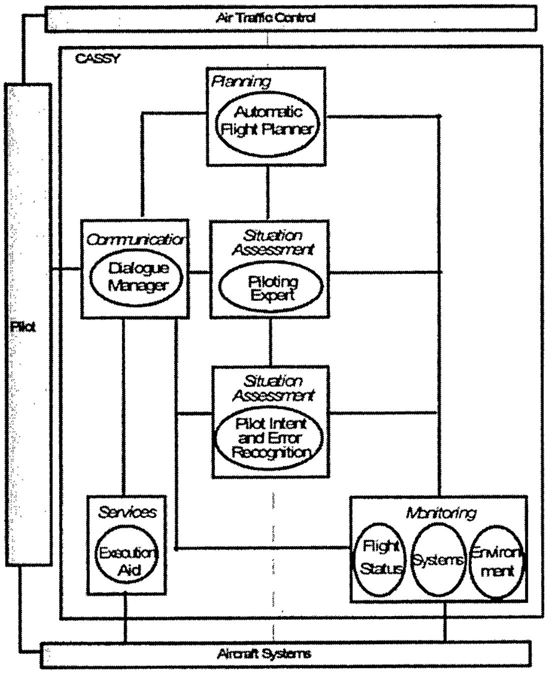
Picture 3: functional layout of an CAS

Discrepancies are classified as errors or intents. Errors lead to warnings with explicit description what to do, intents are automatically incorporated into a flightplan update(e.g. go around: creating a new plan based on the go around procedure).

Conflicts between flightplan and situation (e.g. thunderstorm) are detected, conflict solutions (e.g. a new flightplan) generated and proposed to the pilot.
Man/machine interaction is carried out via a flight plan situation display, speech output and speech recognition with a dynamic, context-dependend speech model (Dialogue Manager [Gerlach 93]) .

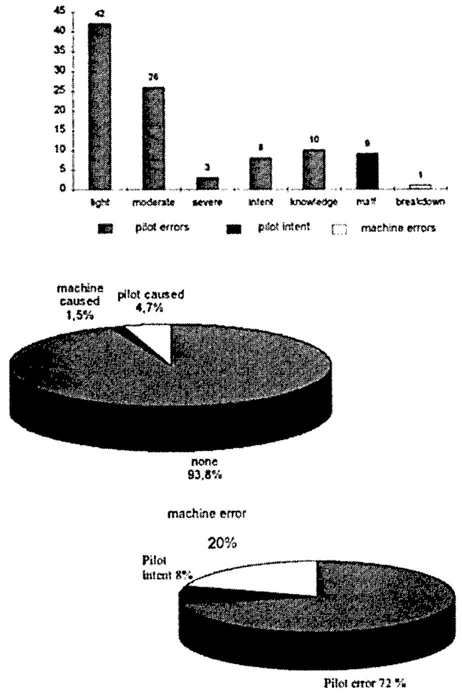
This system was tested with airline pilots in flight simulators and inflight (ATTAS test aircraft of the DLR) in 1994.
The results were very promising: For 94 % of the flight time pilot and cognitive assistant system had the same situation assessment.

The discrepancies detected by CASSY led to alerts. Most of the alerts were caused by pilot errors. False alerts were recognized as such by the pilots and disappeared after completion of a flight manoeuvre.



Picture 4: CASSY

An important result was the symbiotic effect that could be observed: Each error whether caused by CASSY or the pilot was detected by one or the other. Situation awareness of man and machine complemented each other.



Picuture 5: Causes for discrepancies during flight trials of CASSY

3.3 CAMA, Crew Assistant Military Aircraft

Based on CASSY, the Crew Assistant Military Aircraft (CAMA) is under development since 1995 in close cooperation with ESG, DASA and DLR .

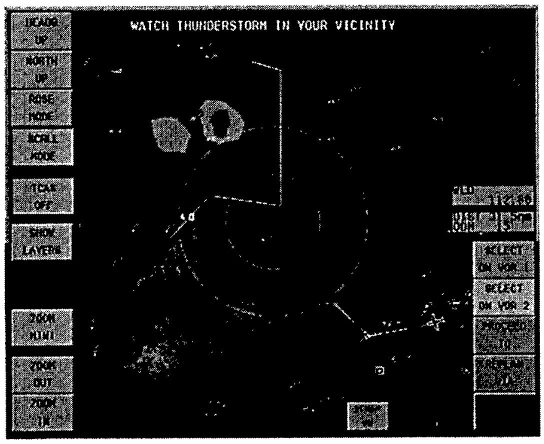
As military transport missions also include IFR flight phases, the complete IFR functionality of CASSY is incorporated into CAMA.

A main feature of the CAMA architecture is a „Central Situation Representation" (CSR), which is fed with data by sensors and situation analysis modules like a Tactical Situation Interpreter or a Terrain Interpreter.

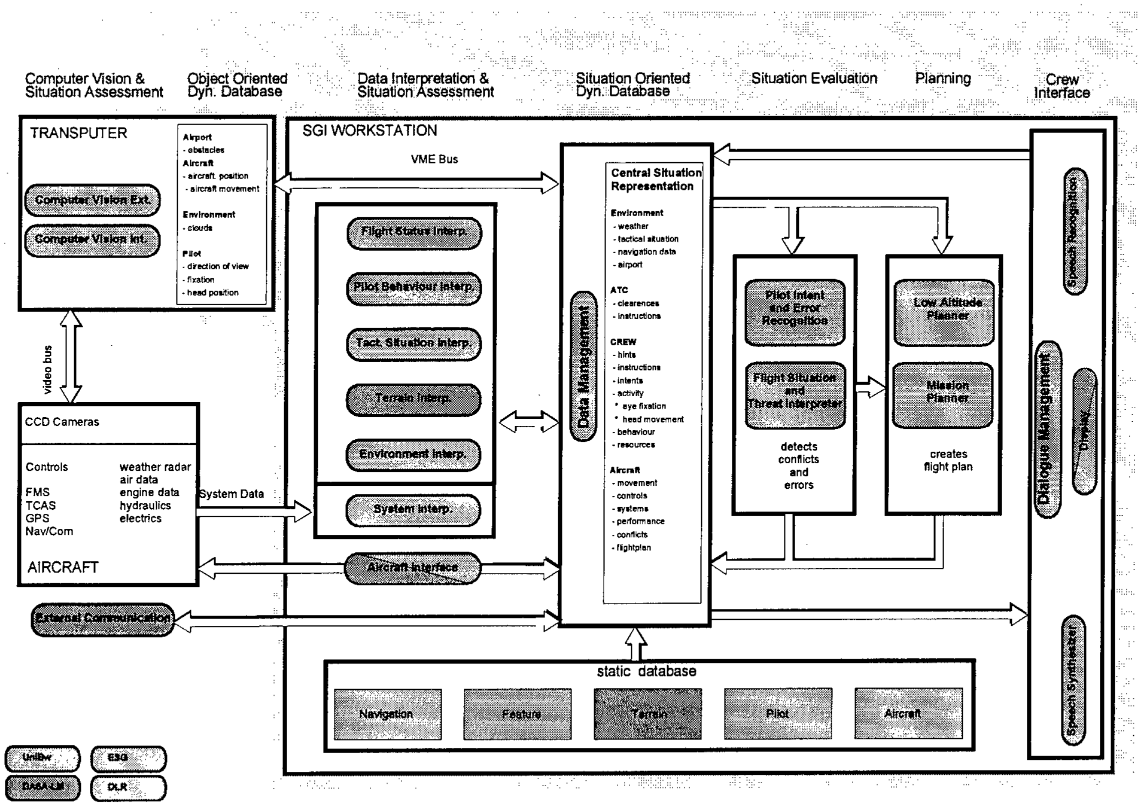
Situation data is the basis to establish the expected individual pilot behavior (Piloting Expert using Petri nets and Cased Based Reasoning, [Stütz 96]), which is compared with actual behavior by a fuzzy-neuro Intent and Error recognition [Strohal 1995].

If conflicts are detected (Flight Situation and Threat Interpreter [Walsdorf 97], problem solving modules like a Low Level Flight Planner [Schulte et al 97] are started. Problem solutions are communicated between CAMA and the pilot through a Dialogue Manager.

This system was tested in a flight simulator with German Airforce pilots, additional simulator trials will take place in spring 98, flight trials are scheduled for early 2000.


WATCH THUNDERSTORM IN YOUR VICINITY

Picture 6: Interactive Nav-Display CAMA



Picture 7: CAMA

4. The CAMA Dialogue Manager

CAMA offers a lot of relevant information and intelligent features. Whatever problem with informational overload or system behavior described in chapter 2 might occur, the impact of this problem will pop up at the communication interface between operator and assistant system. CAMA tries to avoid these communication problems through a specialized modul, the Dialogue Manager (DM).

Examples of information transfer from pilot to CAMA are.:

- requests for new flight plan proposals
- activation of proposals
- activation of actions that are related to warnings
- requests (e.g. for information about airports, navigation aids, enemy weapon systems)
- autopilot operations
- configuration of the man/machine interface

Examples of information transfer from CAMA to the pilots are:

- presentation of proposals
- presentation of the situation (weather, tactical situation, terrain, nav-aids, airports ...)
- alerts about possible conflicts
- warnings with explicit reaction
- messages in reply to requests
- acknowledgement of speech input
- presentation of complex actions (briefing)

The DM manages the following interface components:

- a true moving map display (textured) with alternatively selectable digital map (1.250000 low level flying chart), terrain (DTED), threat, symbolic and textual presentation
- a secondary display for e.g. alphanumeric flight plan, ATIS, overview of the situation, briefing
- speech output system which codes urgency of messages into different voices
- context sensitive speech input system

How can an intelligent dialogue management cope with the challenge of informational overload, high system complexity and the limitations of human information processing?

4.1 Multimodality

Every message from CAMA to the pilot is done simultaniously by speech output and textual feedback.

Whatever information transfer is necessary from the pilot to CAMA, it can be done alternatively through speech or manual input. This gives the pilot the chance to choose the appropriate channel suitable to the actual situation. In head up situations, e.g. final approach he ought to keep his eyes outside and hands on the controls, in head down situations, e.g. a complex planning situation, he should be able to act on that what he already looks at.

4.2 Object-oriented selection

By finger pointing (touch screen) on map symbols the pilot gets presented information and action possibilities related to that symbol. Thus, selecting a VOR can be done by just two actions: touch the VOR symbol on the nav map, secondly touch the upcoming „Select on VOR 1“ button. This can be of advantage because of two aspects:

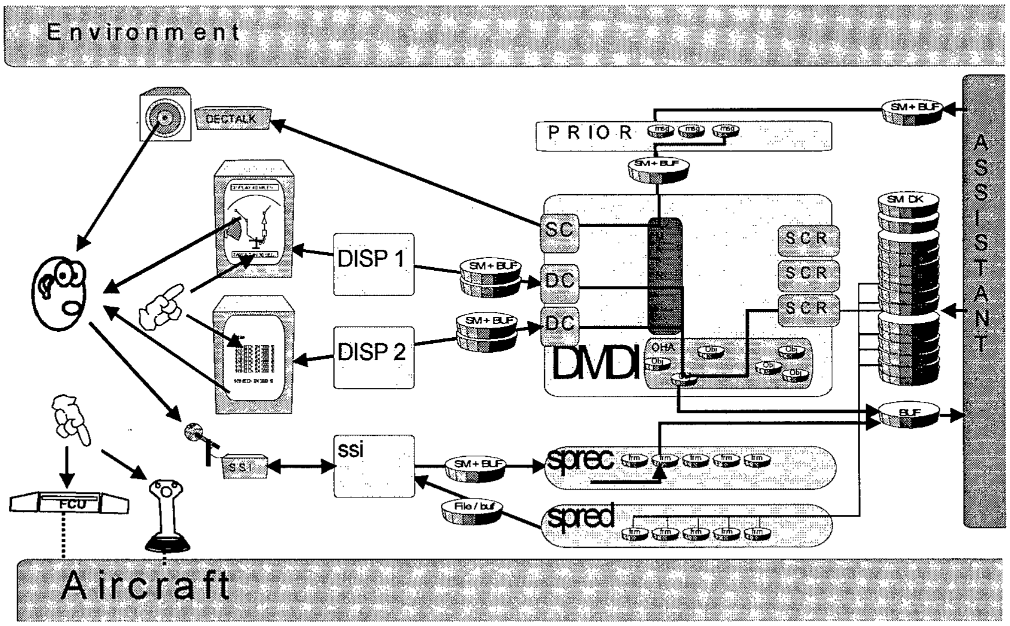
- less load on short term memory („What's the code for this station?“). This increases stress robustness.
- the operator manipulates a situational element directly in its context and mental representation (ecological) it is used later on(e.g. the purpose of a VOR is navigation, the mental representation for navigational aspects is more similar to a 2D-map than to a frequency code). Thus, benefits for situation awareness can be expected.

4.3 Priorisation

In situations with more than one CAMA output these messages are ranked with respect to situation-dependent urgency/importance, then queued and presented with enough time for the pilot to react. In flight phases with high workload (e.g. drop) certain messages with lower urgency/importance might be queued until interaction resources are available.

4.4 Preselection of information

As there is much more information in the system than can be simultaniously handled by the pilot, emphasis is given for the presentation of that information which is important and relevant in that actual situation. This cannot be done without a situational understanding of CAMA.
One example: If the display range exceeds a certain value, only those airports are displayed which belong to the class of potential alternates. On a secondary display page these airports are listed in order of importance.



Picture 8: Dialogue Manager CAMA

4.5 Situation-dependent automatic interface configuration

In a few situations CAMA can change display range, mode or pages automatically. This is neccessary, for instance, in flight phases with high workload (e.g. display range drop, landing) or short reaction time (e.g. TCAS resolution advisory). In order not to escalate interface complexity and thereby increase the risk of automation surprises through automatic changes of the interface configuration must be handled with care.

4.6 Implicit adaptation

High workload situations and the occurence of severe conflicts always give rise to the following question: Should the system automatically activate conflict resolutions like a new plan and should it act directly to the aircraft controls? How can the risk associated with this approach (see also [Verwey 1990]) be avoided?
One possible answer for a lot of situations with resource problems is implicit adaptation: Through a certain style of dialogue the system offers different ways for the operator how to react to a conflict. This includes a very brief and easy one that can even be done in extreme situations. Thereby, the pilot has got the means to manage his own resources across all situations.
One example: A change of the tactical situation (e.g. a new SAM detected) can result in a severe conflict, which can be solved through a new flight plan update (e.g. a new minimum risk route to a new corridor). This is how it works with CAMA: CAMA

- tells the pilot that the situation has changed
- tells the pilot that there is a conflict with this situation change.
- tries to figure out a solution
- offers a solution „Replan via corridor TK05?“
- pilot can accept this solution just by saying „roger do it“
- or, if he has got enough resources, he can analyse the situation himself and find other solutions (which could be better for his own situation awareness)

5. Outlook: caSBAro and interaction resources

How do we proceed with our assistance system in the next future?
Besides further knowledge refinement and functional exploitation we try to improve human factors evaluation and the systems understanding of the pilots resources, an approach that will be explained in further details:

Situation and pilots behavior are strongly connected with each other, only with consideration of both there is the chance to analyse or model operators behavior:

„Now if there is such a thing as behavior demanded by a sitation, and if a subject exhibits it, then his behavior tells more about the task environment than about him. [...] If we put him in a different situation, he would behave differently“

[Newell & Simon, 1972]

There are a lot of different methods for analysing human behavior. [Singleton 1974] distinguishes between

- **task analysis**: tasks / functions are deduced top down deduced from system consideration
- **job analysis**: actual behavior is analysed, more abstract models of behavior are composed from real data.

Related to these basic methods are **activity analysis** [Breuer, Rohmert 1993], which analyses parameters like response time or strength, or **time line analysis**, which compares the task execution time with a temporal task model, thereby detects resource bottlenecks.
Primary Task, **Secondary Task** and **Embedded Secondary Task** measurement provide data about mental states (e.g. workload) by correlating performance of different subtasks [O'Donnell, Eggemeier 1986].
Methods that decribe behavior from situation parameters can be called **situation taxonomy .**

As about 70% of all human perception is done through the visual channel, **eye movement analysis** is an important method of job analysis [Chase 86].
Physiological measurement or subjective methods (questionaires / ratings) can deliver additional information about mental states.

Each of these methods has benefits in some ways and weaknesses in others. Often only a combination of methods can deliver reliable information.

caSBAro (computer aided situation and behavior analysis replay / online) is a base tool for combining behavior / job analysis methods with respect to operator - assistant system interaction.

caSBAro can

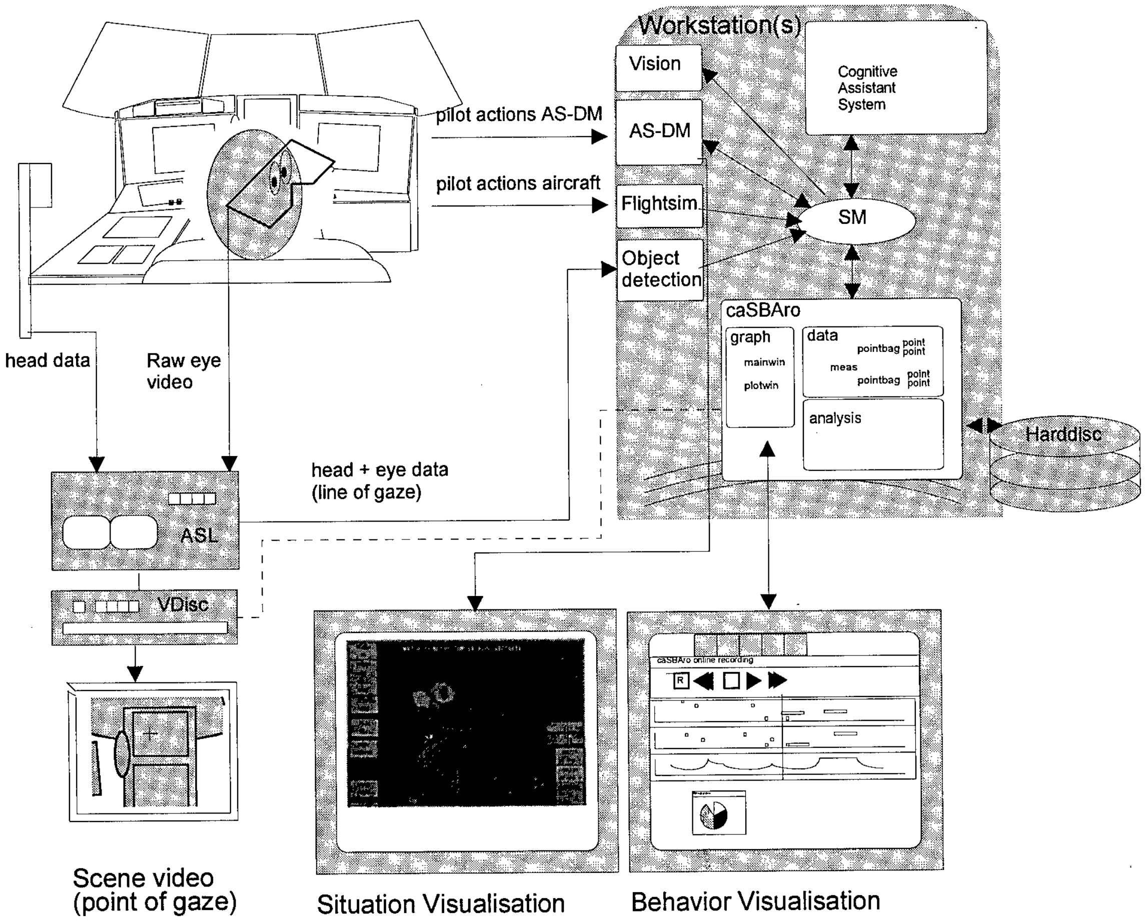
- **record**
  - situation parameter
  - operator interactions including eye movement
  - behavior of the assistant system
- **visualize** this data (replay / online)
- **replay** this data in realtime (the simulator testbed and assistant system behaves as if there is acting a real operator, so giving operators and designer the chance for direct perception of problems)
- **analyse** this data (replay / online)

The requirement of realtime visualisation and analysis is motivated by two aspects:

- analysis toolbricks (e.g. for mental resources) can be developed and pretested with replay data. As these pretests are in realtime, the toolbricks can be plugged into the assistant system without further problems.
- Most problems that can occur in man/machine interaction are strongly dependent on time. However, since we have to decompose man/machine interaction for analysis, e.g. to handle it by use of computers and mathematics, eventualy they have to be brought back to a more holistic and intuitive representation. Bringing analysis back into the timely context is often the only chance to illustrate (interface) these problems to operators (e.g. problems with operators resource management) or system designers (e.g. problem with a time delay of a support function):

"Science, whatever be its ultimate developments, has its origin in techniques, in arts and crafts.... Science arises in contact with things, it is dependent on the evidence of the senses, and however far it seems to move from them, must always come back to them."

[Farrington 1949]



Picture 9: caSBAro

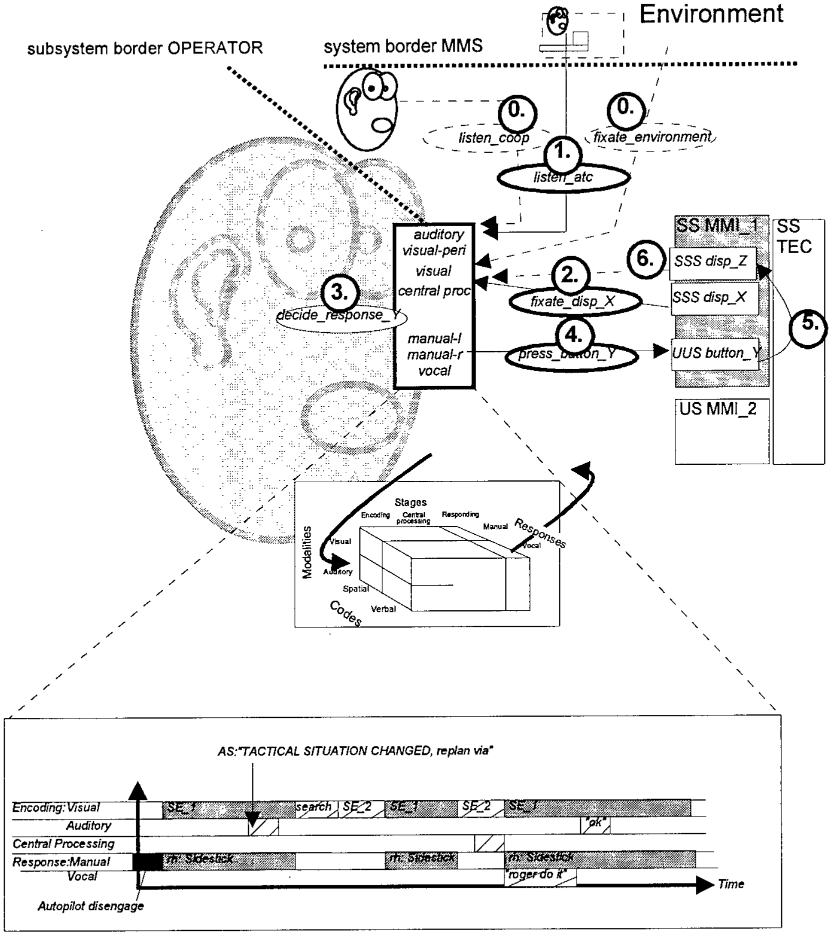
One benefit from caSBAro is to give the system designer and the assistant system a better understanding of what's going on with the operator's interaction resources. This provides the basis to adapt the interaction to the operator's needs.
This adaptation can be done online or offline, by the operator or by the system / system designer. Detailed questions have to be answered to direct the effort efficiently (Picture 10).

Considering these potential targets of a resource diagnosis, it is obvious we need

- a sufficient level of detail to identify problems with single MMI-subsystems / functions / resource channels.
- a perspective that is wide enough to identify connections and timeline of interaction.

	Operator	System / Designer
offline	**train** What task? Or understanding of the system? Or procedures? Or own resource management ?	**redesign** What feature? What behavior? Or change of function allocation → which function?
online	**adapt** own resource management **reallocate** function	**don't disturb** ORM **support** ORM **offer** to reallocate reallocate function

Picture 10: What can be done when resource problems are detected?



1. 2. 4.: basic interactions that can be observed

0.: basic interaction that trie to use a channel already occupied

3: Central processing that cannot be observed but can be estimated if the underlying interaction pattern is already detected

three representations of mental resources:

- middle: multiple resource model [Wickens 92]
- top: simplified and modified
- botton: resource time line, pattern already classified , enabling event „AS TACTICAL SITUATION CHANGED"

Picture 11 : diagnose of interaction resources

Thus, the basic bricks a resource diagnosis can be build up with are:

- **basic interaction** (a discret event between an interaction resource of the operator and a subsystem outside of the operator) as smallest interaction element satisfying the demanded level of detail

- **interaction pattern** (grouping of basic interactions that are somehow related to each other) as the next higher interpretation level.

- **Situation knowledge**

Most of these basic interactions can be observed (e.g. fixation of a subsystem „display“ by eye movement measurement, pressing a button by keystroke recording).
Knowledge aquisition about how basic interactions are related to each other and to the situation can be done through direct (recorded) observation.

Based on this knowledge, typical interaction patterns can be detected or estimated by observation of:

- interactions already happened
- situation
- enabling events of the technical system

What we should not expext from this method are detailed and reliable results for all kinds of situations and interactions (complexity / variability of interaction patterns might be exploding if not sufficiently constrained by the situation, compromise between level of detail, reliability and general usability)

What we can expect from this method:

- It increases the knowledge of the machine / the system designer / the operator about the interaction through direct observation, which is condensed bottom up into more abstract knowledge, additional to top down decomposition (e.g. situation / goals / function / task), so reducing the risk of statements, that are correct but have nothing to do with what really happens.

- It helps to diagnose problems in man-machine-interaction in predefined situations, due to insuffizient operator resource management, unsuffizient mental model of the machine, poor pratice or poor system design
    - with interference between interaction patterns (e.g. how does the interaction with a specific technical subsystem we want to validate fit into the interaction with other partners, e.g. ATC)
    - within a specific interaction pattern (is it too long, is it interuptable, where are the risks for errors or slips, how does the operator perform?)
    - within a specific basic interaction (how long does it take?)
    - with a specific technical subsystem

- It guides to solution for problems observed through a detail level that is high enough.

- It supports the realisation of a problem solution through high face validity.

- It diagnoses resource problems online and delivers data that can be applied by the assistant system to adapt the future interaction.

First results of this concept are expected in early 99.

5. Conclusion

Technical progress with more information available and better technical features offers the potential to improve mission management systems dramatically. However technical progress that does not respect the excellencies of human information processing and does not take into account its limitations, is exposed to performance drawbacks or even degradation of safety.

Cognitive Assistant Systems do not solve all problems that can occur with man/machine interaction and mission systems, but with knowledge about the task and the operator, with situation assessment in parallel by the machine, with automatic conflict detection, conflict resolution and intelligent interfacing these systems yield great potentials to transform technical progress into real operational benefits for future man machine systems.

CAMA is a prototype of a cognitive assistant system for the domain of military transport aircraft. It shows that **assistant systems can be built** and implemented into operational systems in the near future.

References

Baron 1984 Baron, S: *A control theoretic approach to modelling human supervisory control of dynamic systems,* In: W.B. Rouse Advances in man-machine systems research, 1, JAI Press, Greewich, CT 1984

Billings 1991 Billings, C. E.: *Human Centered Aircraft Automation: A Concept and Guidelines,* NASA Ames Research Center Technical Memorandum 103885, Moffet Field CA, 1991

Breuer, Rohmert 1993 Breuer, B., Rohmert, W.: *Driver Behaviour and Strain with special regard to the itinerary*, Prometheus Report Phase III, Technische Hochschule Darmstadt, 1993

Chase 1986 Chase, William G.: *Visual Information Processing*, In: Handbook of Perception and Human Performance, Vol II, Wiley, New York, 1986

Chu & Rouse 1979 Chu, Y.-Y., Rouse, W.B.: *Adaptive Allocation of decisionmaking responibility between human and computer in mulitask situations*, In: Proceedings IEEE Trans. Systems, Man, Cybernetics, Vol. SMC-9, 1979

Eysenck 1993 Eyseck, M. W.: *Principles of Cognitive Psychology*, Lawrence Erlbaum, Hove, UK, 1993

Farrington 1949 Farrington, B.: Greek Science, Penguin Harmondsworth, 1949

FAA 1996 *The Interfaces Between Flightcrews and Modern Flight Deck Systems*, Federal Aviation Administration, 1996.

Gerlach 1993 Gerlach, M., Onken, R.: *A Dialogue Manager as Interface between Pilots and a Pilot Assistant System*, In: HCI 1993, Orlando, 1993

Newell, Simon 1972 Newell, A.; Simon, H.A.: *Human Problem Solving*, Prentice Hall, Englewood Cliffs, 1972

O'Donnell, Eggemeier 1986 O'Donnell, R. D., Eggemeier, T.: *Workload Assessment Methodology*, In: Handbook of Perception and Human Performance II/42, Wiley, New York, 1986

Onken 1997 Onken, R.:*The assessment of situation awareness and workload for certification purposes* In: Proceedings of the European Workshop to Develop Human Factors Guidelines for Flightdeck Certification, London, 1997

Prevot et al 1995 Prévot T.; Gerlach M.; Ruckdeschel W.; Wittig T.; Onken R.: *Evaluation of Intelligent On-Board Pilot Assistance In-Flight Field Trials.* IFAC Man-Machine Systems '95, Cambridge, MA, 1995

Rasmussen 1983 Rasmussen, J.: *Skills, rules and knowledge, signals, signs, and symbols, and other distinctions in human performance models*, IEEE Trans. Systems, Man, Cybernetics Vol. SMC-13, 1983

Rasmussen et. al. 1994 Rasmussen, J.; Pejtersen, A. M.; Goodstein, L. P.: *Cognitive Systems Engineering*, Wileys, Roskilde, Denmark, 1994

Rauterberg 1996 Rauterberg, M.: *A Petri Net Based Analysing and Modelling Tool Kit for Logfiles in Human Computer Interaction*, In: Proceedings CSEPC, Kyoto 1996

Schulte et al 1997 Schulte, A.; Klöckner W.: *Perspectives of Crew Assistance in Military Aircraft through Visualizing, Planning and Decision Aiding Functions*, In:Proceedings AGARD Mission Systems Panel, 6th Symposium, Istanbul, 1996

Singleton 1974 Singleton, W.T.: *Man-Machine Systems*, Penguin, Harmondsworth, UK, 1974

Stütz et al 1997 Stütz, P.; Onken, R.: *Adaptive Pilot Modeling within Cockpit Crew Assistance*, HCI, San Francisco, 1997

Strohal et al 1997 Strohal, M.; Onken, R.: *The Crew Assistant Military Aircraft (CAMA*). HCI, San Francisco, 1997

Taylor et. al. 1989 Taylor, M.M., Neel F., Bouwhuis: *The structure of multimodal dialogue*, Elsevier Science Publishers, North-Holland, 1989

Verwey 1990 Verwey, W.B.: *Adaptable Driver-Car Interfacing*, TNO report IZF 1990 B-3, Soesterberg, NL 1990

Wickens 1992 Wickens, C. D.: *Engineering Psychology and Human Performance*, HarperCollins Publishers, New York, 1992

Wickens 1996 Wickens, C. D.: *Attention and Situation Awareness*, AGARD Workshop, University of armed Forces München, 1996

Wioland, Amalberti 1996 Wioland, L.; Amalberti, R.: *When errors serve safety: toward a model of ecological safety*, In: Proceedings CSEPC , Kyoto 1996

Woods 1984 Woods, D.D.: *Visual momentum: A concept to improve the cognitive coupling of person and computer*, Int Journal Man-Machine Studies Vol. 21, 1984

Walsdorf et al 97 A.Walsdorf, R.Onken, et al.: *CAMA - the Crew Assistant Military Aircraft,* Workshop "The Human Electronic Crew: The Right Stuff ?", September 1997, Kreuth, GE

MACHINE INTELLIGENCE AS APPLIED TO FUTURE AUTONOMOUS TACTICAL SYSTEMS

Uwe Krogmann
Bodenseewerk Gerätetechnik GmbH
Postfach 10 11 55
88641 Überlingen

1 INTRODUCTION

Tactical Systems are implemented as Integrated Mission Systems (IMS) such as air- and space defence systems. Key elements of IMS are platforms with sensors and effectors, ground-based components with communication, command and control etc. The development, procurement and utilization of defense systems will in future be strongly influenced by the affordability issue. A considerable potential for future cost reduction is seen in the extended use of autonomous systems as part of IMS. The key notion of "autonomy" is intimately connected with advances in information technology.

In this context and looking at the title of this paper, the following question arises immediately: What is computational, machine or more generally artificial intelligence? In relation to the issues and topics treated here, the following answer shall be given.

- Systems/units **have no artificial intelligence** if a program/software "injects" them with what they have to do and how they have to react to certain pre-specified situations.
- Systems/units **have artificial intelligence** if their "creator" has given them a structure - not only a program - allowing them to organize themselves, to learn and to adapt themselves to changing situations.

Thus intelligent structures must be able to comprehend, learn and reason.

Moreover, the next important question suggests itself: Why computational intelligence?

For reasons of human limitations in more demanding dynamic scenarios and in the operation of complex, highly integrated systems, there is the necessity for extended automation of functions on higher levels such as mission management and control.

Furthermore, the implementation of intelligent functions on lower levels such as the fusion and interpretation of sensor data, multifunctional use of sensor information and advanced nonlinear learning control become inevitable.

Improved effectiveness in diagnostics and maintenance of systems, subsystems and modules can be achieved through the application of computational and machine intelligence (CMI).

Last but not least CMI will be the most important prerequisite for the emergence of future autonomous systems with the ability fo function as an independent unit or system over an extended period of time, performing a variety of actions necessary to achieve predesignated objectives while responding to stimuli produced by integrally contained sensors.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

2 AUTONOMOUS SYSTEMS

An autonomous system must be provided with the ability to generate goal directed behavior patterns in real world scenarios. The following characteristics are therefore typical of an autonomous, behavior-oriented system:

- An "environment" (real world) is allocated to the system
- There is an interaction between the system and the environment via input and output information and possibly output actions
- The interactions of the system are concentrated on performing tasks within the environment according to a goal-directed behavior, with the system adapting to changes of the environment.

Behavior in this context means a suitable, goal-directed sequence of actions which, as a whole, represent a behavioral pattern.

The interaction of the systems with the surrounding world (Fig. 1) can be decomposed into the following elements of a recognize-act-cycle (or stimulus-response-cycle).

- Recognize the actual state of the world and compare it with the desired state (which corresponds to the goal of the interaction). (MONITORING)
- Analyse the deviations of actual and desired state. (DIAGNOSIS)
- Think about actions to modify the state of the world (PLAN GENERATION)
- Decide the necessary actions to reach the desired state (PLAN SELECTION)
- Take the necessary actions to change the state of the world (PLAN EXECUTION)

To perform these functions, first of-all appropriate sensor and effector systems must be provided.

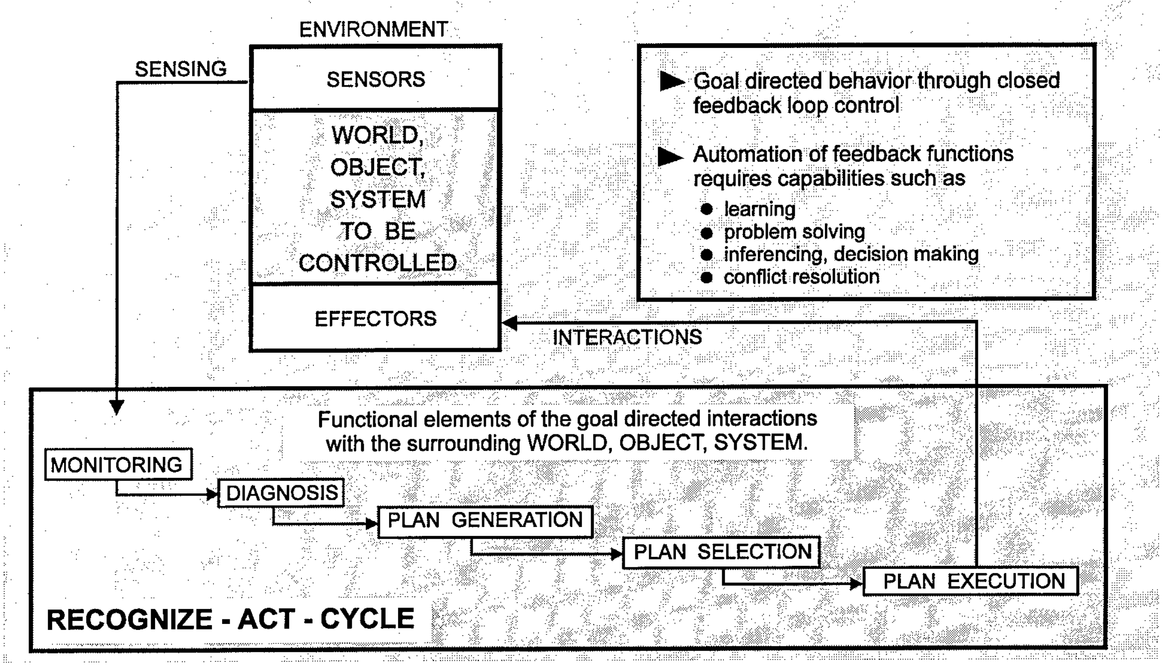

ENVIRONMENT
SENSING
SENSORS
WORLD, OBJECT, SYSTEM TO BE CONTROLLED
EFFECTORS
INTERACTIONS
Goal directed behavior through closed feedback loop control
Automation of feedback functions requires capabilities such as
learning
problem solving
inferencing, decision making
conflict resolution
Functional elements of the goal directed interactions with the surrounding WORLD, OBJECT, SYSTEM.
MONITORING
DIAGNOSIS
PLAN GENERATION
PLAN SELECTION
PLAN EXECUTION
RECOGNIZE - ACT - CYCLE

Figure 1: Interaction autonomous system - real world

The automation of these feedback functions requires capabilities such as learning, problem solving, inferencing and decision making as well as conflict resolution. Therefore, in the case of unmanned autonomous systems information processing means must be incorporated that apply knowledgebased machine intelligence techniques to perform the tasks mentioned. In particular this requires tha acquisition, encoding, storage, processing, recall of knowledge.

The goal-directed interaction of the system with real world objects is made possible through the technical implementation of the endomorphic system concept as represented in Fig. 2 [1].

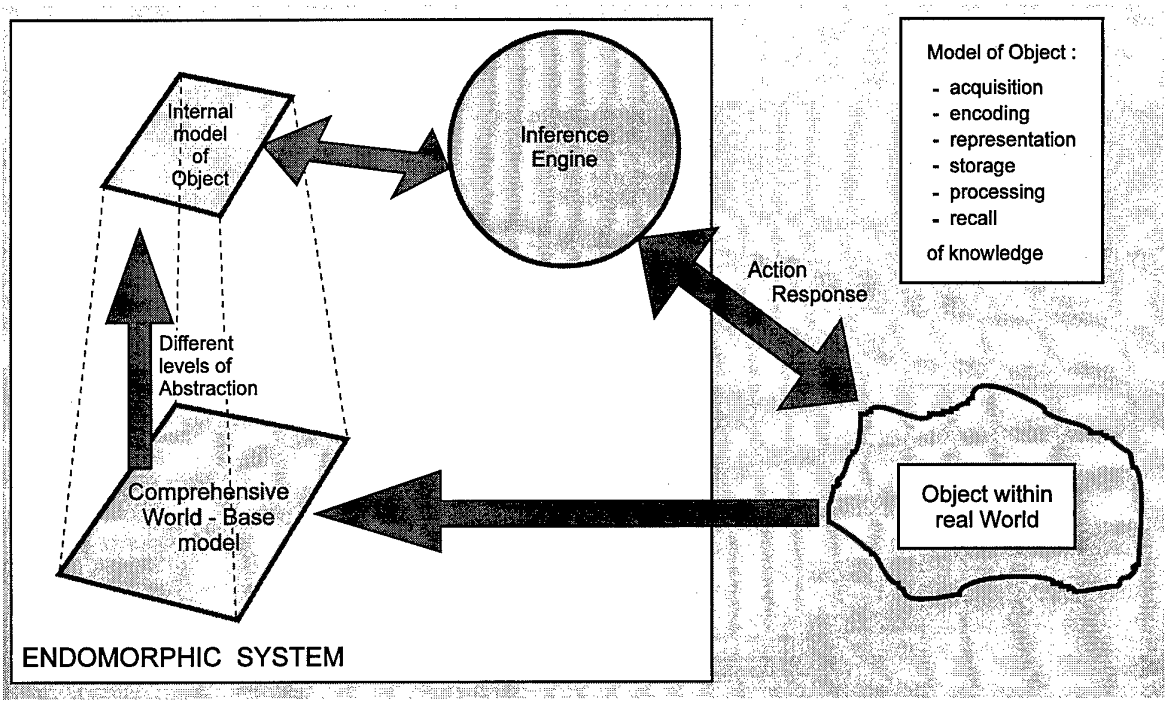


Figure 2: Endomorphic system concept

This is accomplished by establishing an endomorphism between real-world objects. The inference engine inquires at its internal models for necessary information prerequisite to goal-directed interaction with the real-world objects. The applied internal model is abstracted from a comprehensive model of the world and object. The Artificial Intelligence (AI) element of the system comprises a domain-dependent knowledge and data base (real-world modeling) and a domain independent inference mechanism. In general the world-base model comprises different models such as continuous models, discrete-event models, rule-based models. The recognition - act-cycle, involving sensing, planning and execution, can thus invoke different models in an optimum way.

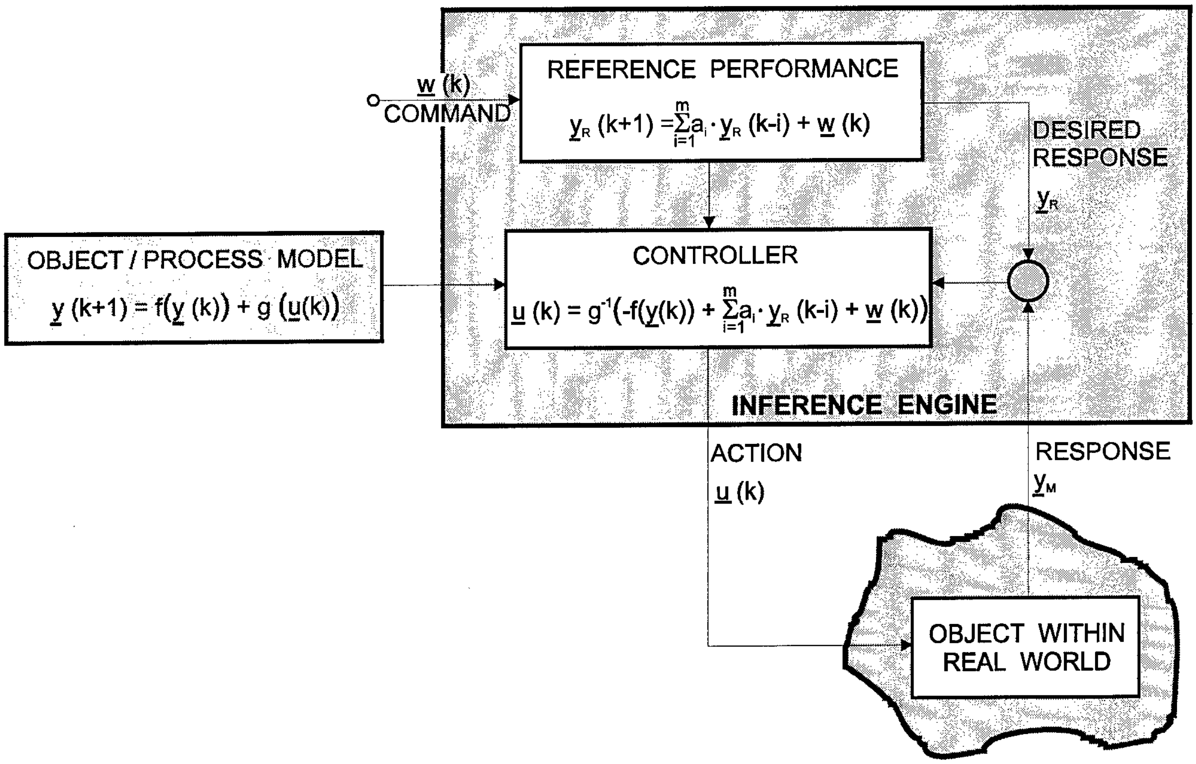


Figure 3: Endomorphic system blockdiagram

A mathematical representation of the endomorphic system structure is shown in Fig. 3 in a simplified discrete time format. It contains a nonlinear model of the controlled object. Moreover a nonlinear controller and - in this case to simplify matters - a linear model to describe the reference performance are incorporated, which together form the inference engine. To implement the goal directed action-response control process, nonlinear functional relations like $f(\cdot)$, $g(\cdot)$, $g^{-1}(\cdot)$ must be mapped. This can be accomplished by the acquisition, encoding, representation, storage, processing and recall of the knowledge, describing the said functional relations.

Although the development of hardware and software for computers of conventional v. Neumann architecture has continued for more than 20 years and the performance of today´s processors is 25.000 times better than in the 1970s, the dynamics of this development is going on as well.

The expected unprecedented advances in computing based on the conventional architecture, where processing is performed sequentially, do not yield the power for knowledge based computional and machine intelligence (CMI) as defined previously in chapter 1.

However, there is a complementary shift from conventional computing techniques, including symbolic AI/KB techniques, to so-called soft computing technologies. The new paradigm is based on modelling the unconscious, cognitive and reflexive function of the biological brain. This is accomplished by massively parallel implementation in networks as compared to program/software based information processing in conventional sequential architectures.

3 SOFT COMPUTING

In contrast to the conventional method, soft computing addresses the pervasive imprecision of the real world. This is obtained by consideration of the tolerances for imprecision, uncertainty and partial truth to achieve tractable, robust and low-cost solutions for complex problems.
Important related computing methodologies and technologies include among others fuzzy logic, neuro-computing, as well as evolutionary and genetic algorithms [2] - [7].

The theory of fuzzy logic provides a mathematical framework to capture the uncertainties associated with human cognitive processes, such as thinking and reasoning. Also, it provides a mathematical morphology to emulate certain perceptual and linguistic attributes associated with human cognition. Fuzzy logic provides an inference morphology that enables approximate human reasoning capabilities for knowledge-based systems. Fuzzy logic/fuzzy control has developed an exact mathematical theory for representing and processing fuzzy terms, data and facts which are relevant in our conscious thinking.

Neural Networks are derived from the idea of imitating brain cells in silicon and interconnecting them to from networks with self-organization capability and learnability. They are modeled on the structures of the unconscious mind. Neurocomputing is a fundamentally new kind of information processing [3]. In contrast to programmed computing, in the application of neural networks the solution is learnt by the network by mapping the mathematical functional relations.

Genetic and evolutionary algorithms represent optimization and machine learning techniques, which initially were inspired by the processes of natural selection and evolutionary genetics [4]. To apply a genetic algorithm (GA) potential solutions are to be coded as strings on chromosomes. The GA is populated with not just one but a population of solutions, i.e. GA search from a population of points rather than from a single point. By repeated iterations a simulated evolution occurs and the population of solutions improves, until a satisfactory result is obtained.

With these techniques a viable step towards intelligent machines can be expected that offer autonomous knowledge acquisition and processing, self-organization and structuring as well as associative rule generation for goal-oriented behavior in rarely predictable scenarios. The new techniques will yield computational and machine intelligence which offers the user the opportunity for cognitive automation of typical „recognition-act cycle“ activities on various functional and operational levels.

The last two decades have witnessed a very strong growth of CMI techniques. These techniques have already been applied to a variety of problems to deliver efficient solutions to the benefits of the user. Certainly there are relationships between CMI and other fields such as those shown on top of Fig. 4. Moreover, numerous disciplines have contributed to the area of soft computing where some are mentioned at the bottom of Fig. 4.

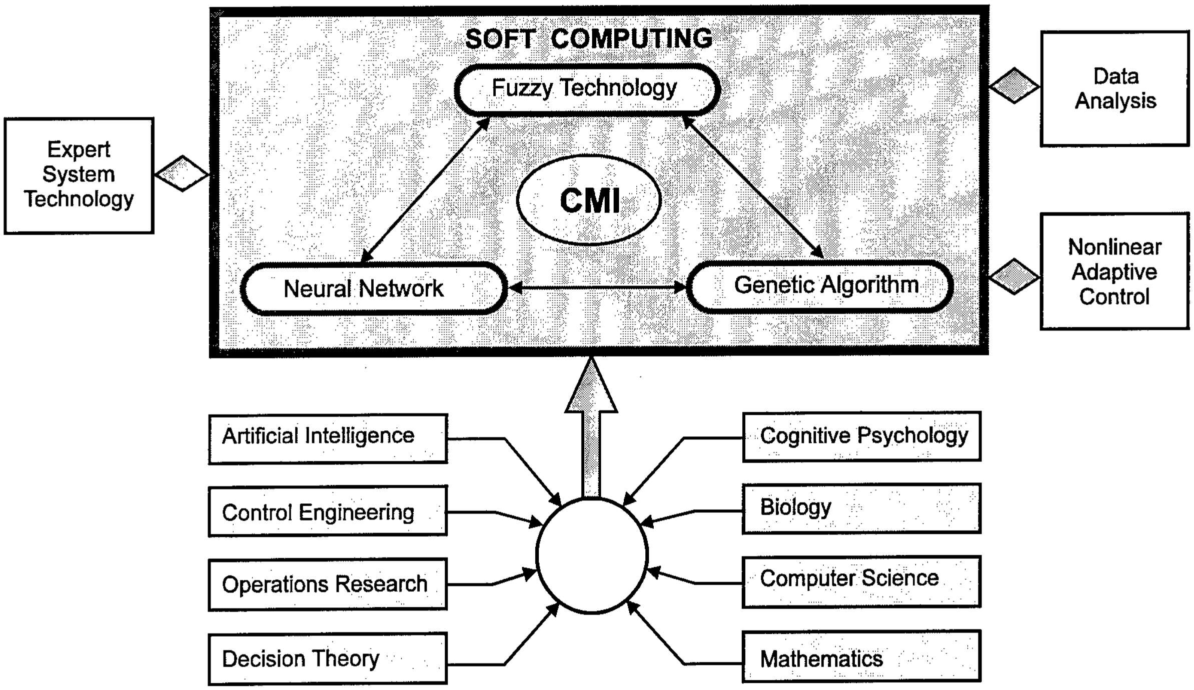


Figure 4: ***Relationships to and contributions from other areas and soft computing/CMI***

Fuzzy and artificial neural network techniques enable the endomorphic modelling of real world objects and scenarios. Taking the approximation of the simple function y = 1-cos(x) as an example, this is visualized in Fig. 5. Apart from conventional analytical techniques it represents new powerful means for knowledge acquisition, representation and processing prerequisite for functional mapping.

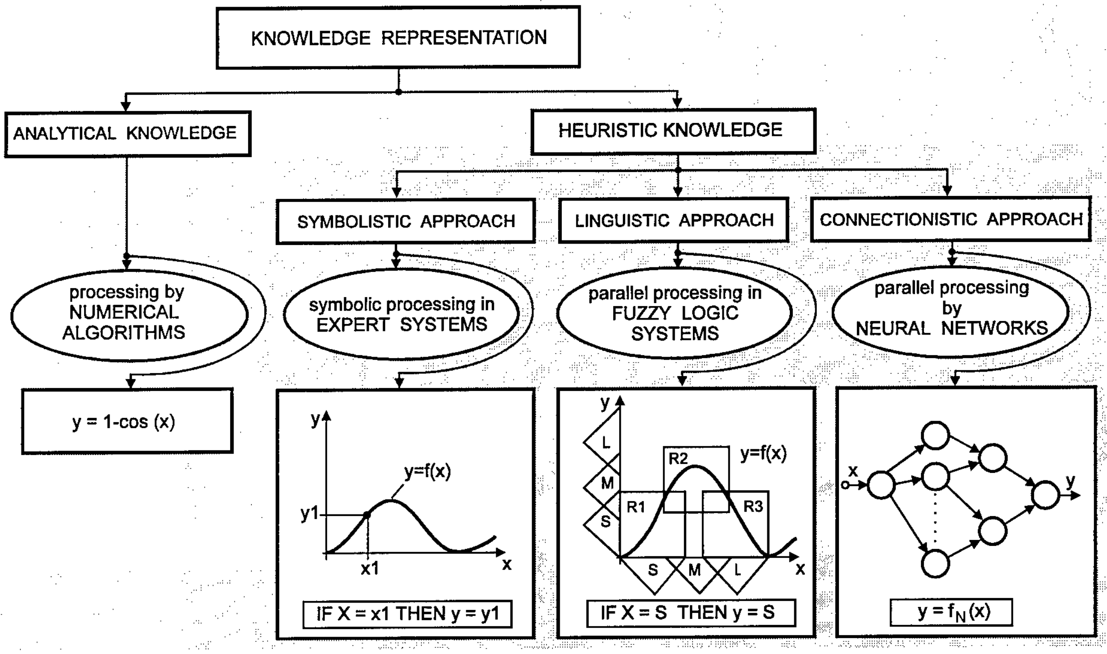
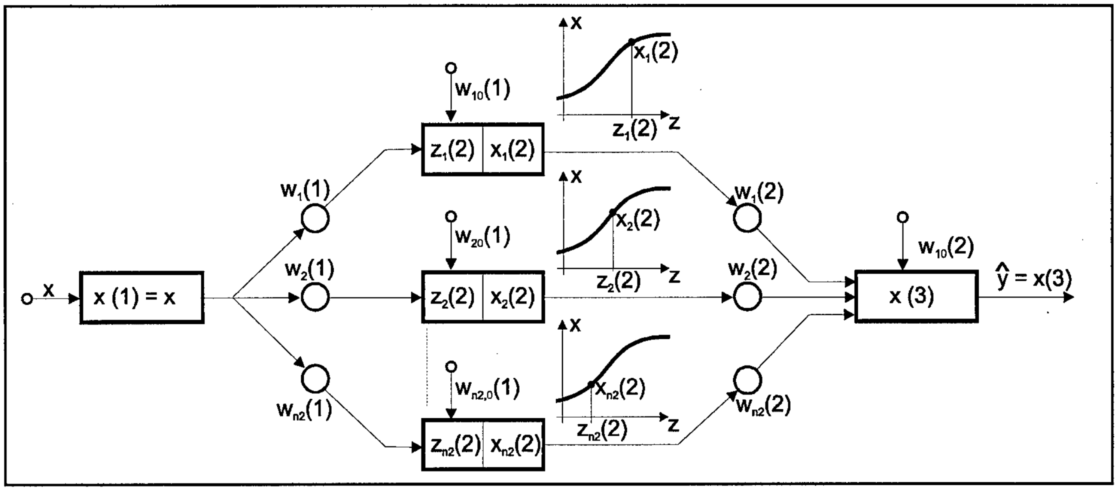


Figure 5: ***Knowledge representation and processing***

With the symbolistic approach crisp If .... Then rules are applied to describe the relationship, whereas the fuzzy representation unilizes fuzzy If .... Then rules relating linguistic labels of fuzzy sets. A neural network learns the functional relationship $f_N : x \rightarrow y$. As can be seen from the simple network in Fig. 6, it mappes a nonlinear regression polynominal. The coefficients (w) are adjusted during learning by processing training data pairs x*, y*.



$$\boxed{\hat{y} = w_{10}(2) + \sum_{i=1}^{n_2} w_i(2) \cdot f\,(w_i(1) \cdot x(1) + w_{io}(1))}$$

Figure 6: Function approximation with a neural network

Together with conventional algorithmic processing, classical expert systems, probabilistic reasoning techniques and evolving chaos-theoretic approaches the techniques treated here enable the implementation of recognize-act cycle functions as shown in Fig 1. Genetic and evolutionary algorithms can be applied to generate and optimize appropriate structures and/or parameters to acquire, encode, represent, store, process and recall knowledge. This yields self-learning control structures for dynamical scenarios that evolve, learn from experience and improve automatically in uncertain environment. Ideally, they can be mechanized by a synergetic complementary integration of fuzzy, neuro and genetic techniques (Fig. 7). Fuzzy logic for decision making and reasoning, neural networks for learning and selforganization and genetic algorithms primarily for task oriented optimization. These soft-computing techniques support the move towards adaptive knowledge based systems which rely heavily on experience rather than on the ability of experts to describe the dynamic, uncertain world perfectly. Thus, soft-computing techniques in conjunction with appropriate system architectures provide the basis for creating the above-mentioned Behavior-Oriented Autonomous Systems. In the following, this will be looked at in somewhat greater detail.

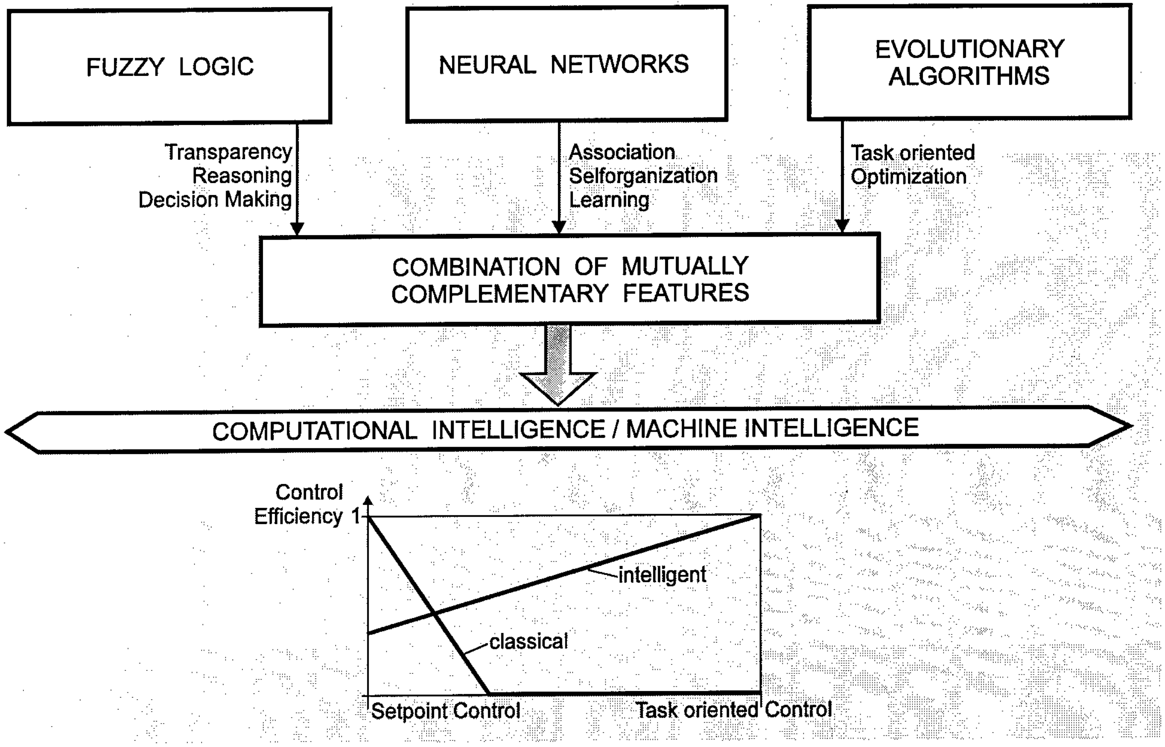


Figure 7: ***Complementary features of technologies***

4 CONCEPTUAL IDEAS FOR AUTONOMOUS SYSTEMS

4.1 System architectures

The viable architecture must represent the organization of the systems functionalities, intelligence, knowledge and capability to behave, to learn, to adapt and to reconfigure in reaction to new situations in order to perform in accordance with its functionalities. Based on fundamentally different philosophies regarding the organisation of intelligence, two different architectures can be basically considered (Fig. 8). With the well known top-down approach as prevalently used to date a hierarchically functional architecture results. It structures the system in a series of levels or layers following the concept of increasing precision with decreasing intelligence when going from top to bottom. Implementation is characterized by the fact that for as many contingencies as possible the allocated system behavior is fixed in top-down programming. This method is "unnatural" and, according to the definition given above, does not lead to artificial intelligence of the system. In fact, the real world is so complex, imprecise and unpredictable that the direct top-down programming of behavioral functions soon becomes almost impossible.

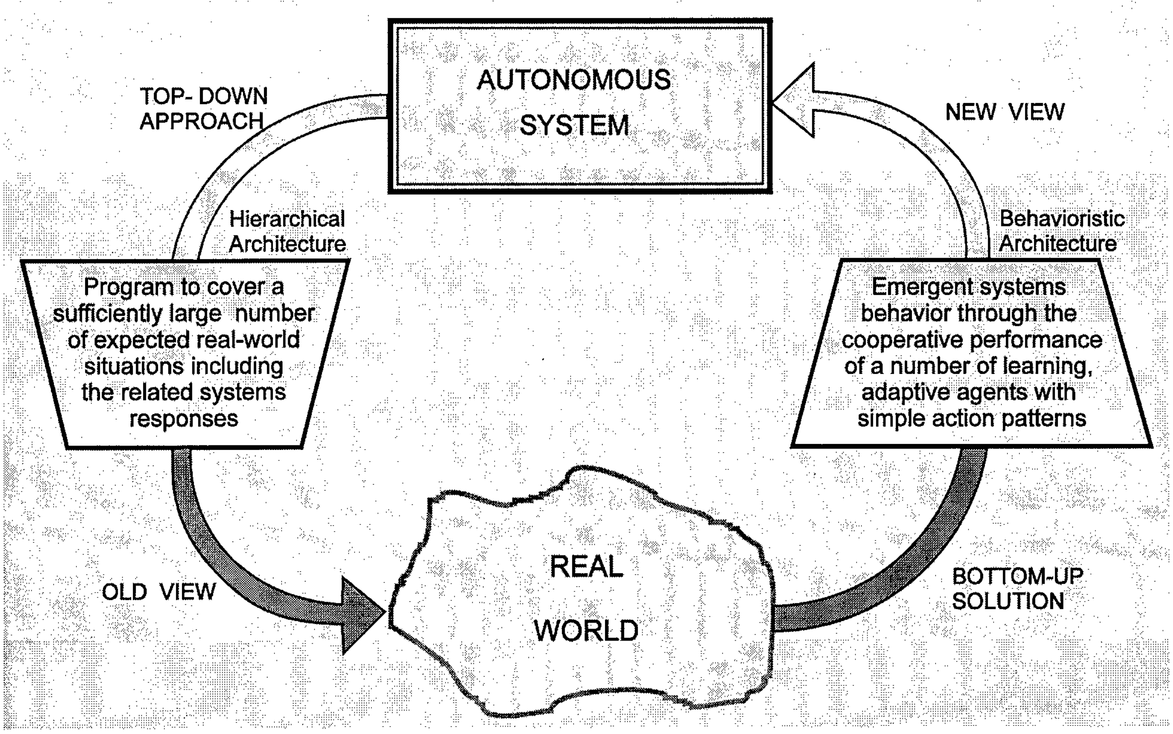
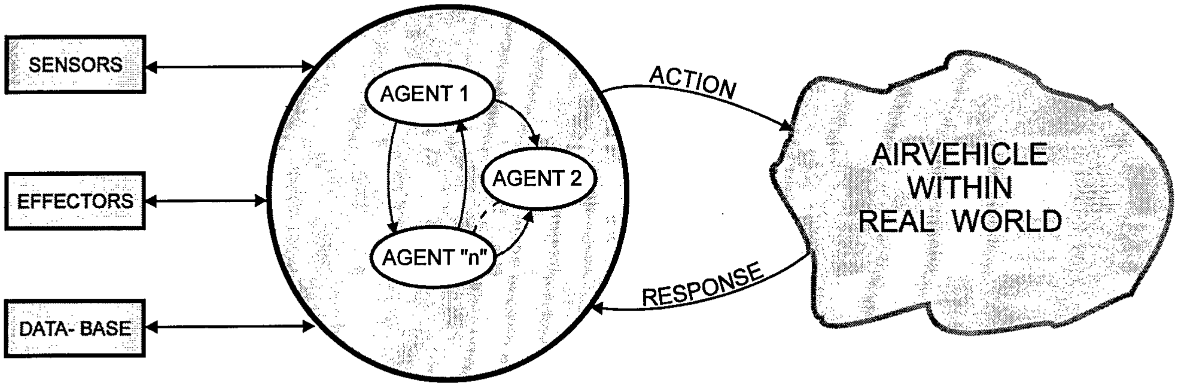

AUTONOMOUS SYSTEM
TOP- DOWN APPROACH
Hierarchical Architecture
Program to cover a sufficiently large number of expected real-world situations including the related systems responses
OLD VIEW
REAL WORLD
NEW VIEW
Behavioristic Architecture
Emergent systems behavior through the cooperative performance of a number of learning, adaptive agents with simple action patterns
BOTTOM-UP SOLUTION

Figure 8: Organization of the systems intelligence

Considerably different from the hierarchical structure is the subsumption architecture. It is based upon building functionality and complexity from a number of simple, parallel, elemental behaviors. It is sometimes called the behaviorist architecture and is based on a buttom-up approach. In this approach, so-called agents are implemented with the most simple action and behavior patterns possible so that the resulting emergent system behavior corresponds to the desired global objective. The system is able to adapt itself to changing situations in the environment by learning. The specific local intelligence of the individual agents, implemented with soft computing techniques, generates a global intelligent behavior of the integrated overall system. Multi-agent systems are very complex and hard to specify in their behavior. Therefore there is the need to endow these systems with the ability to adapt and learn. This learning capability is very important, because otherwise a multi-loop nonlinear control problem of very high order must be solved. The current state of the art precludes a solely analytic solution for such a complex system. For this reason we invoke at the field of biology.

The biological nerve system is the living example for the fact that strongly meshed systems of an extremely high order can adopt stable states. Moreover, without supervised control, these biological systems are able to act purposely and task oriented. By an extensive comprehension of the biological paradigm, the brain, we must try and strive to recognize the regularities which might be of decisive use to us for the stabilization and self-organization of highly integrated complex dynamic systems.

A simplified block diagram of an autonomous airvehicle system based on such a concept of cooperative AI/KB-Agents, is depicted in Fig. 9 [8]. The objective is to implement as many simple agents as possible with the associated behavior pattern, which then make the system act in a flexible, robust and goal-oriented manner in its environment through their additively complementary interaction. To enable the generation of emergent characteristics it must be ensured that the agents can influence each other mutually. Emergent functionality is one of the major fields of research dedicated to behavior-oriented systems.



AGENTS						
SENSOR DATA FUSION	PERCEPTION, SITUATION ASSESSMENT	PLANNING	....	PILOTING	THREAT	COMMUNI-CATION

Figure 9: Endomorphic system representation by agents

Intelligent HW/SW agents will fuse sensor information, monitor critical variables, pilot the vehicle, generate optimized plans, alert operators through communication to problems as they arise and recommend optimized solutions in real time. Response agents capture basic data, communication (forecast and other information) and apply optimization technology to generate new plans based on changed conditions and states. Remote Operators can consider recommended plans in a "what if" manner, i.e. making changes to the agents suggestions or accepting new recommendations. Last, but not least, agents will offer learning capability.

4.2 Design Consideration

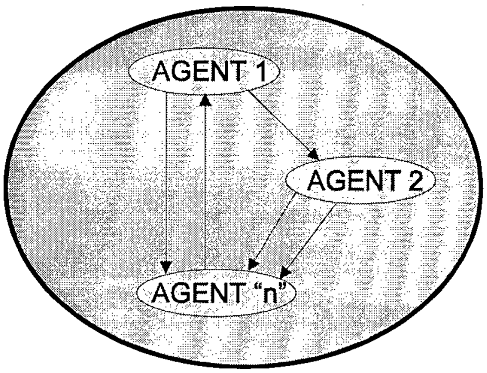
Human or more generally biological beings are not a kind of Bio-Robots, whose behavior is predetermined by their role in a fixed hierarchical structure, as applied in the top-down approach in Fig. 8 [9]. Open hierarchical structures is the general organizational principle of biological life. Hierarchies are essential in biological structures, but they are not fixed. Hierarchies are temporary and change according to present goal and objective. The behavioristic or subsumption architecture of Fig. 8 ia an attempt to map said biological structures.

A computational structure which is expected to bring about the emergence of cognitive capabilities has to feature four important aspects:

- It has to be modular and parallel in order to represent many concurrent processes activated by signals from a number of sensory-motor inputs.
- The individual modules need to be able to learn from experience.
- The interaction between the various modules has to be dynamic in order to guarantee flexibility and plasticity.

- The modules shall integrate themselves into the structure but at the same time try to preserve their autonomy as self-contained independent units.

First of all, as can be seen from Fig. 10, the design of a multi agent system requires the analytical modelling and solution of the multi-dimensional task control problem, where the controlled object is e.g. the situation within the scenario. Assuming, the individual agent can be described as dynamical systems with also dynamical interconnections between the agents, this is a very difficult, expensive, in some cases almost impossible to solve task. Moreover, it does not lead to real AI. It was mentioned already that we need to endow these systems with the ability to learn, where different learning procedures can be applied [3]. Important for autonomous systems are neuronal, genetic and immunitary learning methods [8].



$$\dot{\underline{x}}_1(t) = f_1(\underline{x}_1(t), \underline{u}_1(t))$$

$$\dot{\underline{x}}_2(t) = f_2(\underline{x}_2(t), \underline{u}_2(t))$$

$$\vdots$$

$$\dot{\underline{x}}_n(t) = f_n(\underline{x}_n(t), \underline{u}_n(t))$$

$$\boxed{\dot{\underline{X}} = \underline{f}(\underline{X}, \underline{U})}$$

Figure 10: Multi-agent system

Learning turns out to be a metadynamical procedure, where the learning procedure changes the dynamics $\underline{f}(\cdot)$ of the system. For this reason we are confronted with the stability issues influenced by three dynamical processes

* $\underline{w}$-dynamics (synaptic dynamical system)
* $\underline{f}$-dynamics (activation dynamics system)
* joint $\underline{g}$- and $\underline{f}$-dynamics

Consequently the agents and their interconnection structures as well as the learning procedure must be selected carefully.

4.3 Engineering method

Like in Engineering, it is also an indispensable prerequisite for an autonomous system that it is designed, constructed and trained according to a strict methodical approach. Fig. 11 shows such an approach in a very simplified form from today's technological point of view.

It starts with the description of the physical system, its application, the initial environment, and the behavior requirements, with the latter being usually informally stated in natural language. The following behavior analysis is one of the major tasks. This step involves the decomposition of the target behavior in simple behavioral components and their interaction. Part of the specification is the architecture of the intelligent control system. It is the second key point during the engineering process. With the specification all information is available to design, implement and verify a nascent system, which is endowed with all its hardware and software components, however, prior to any training.

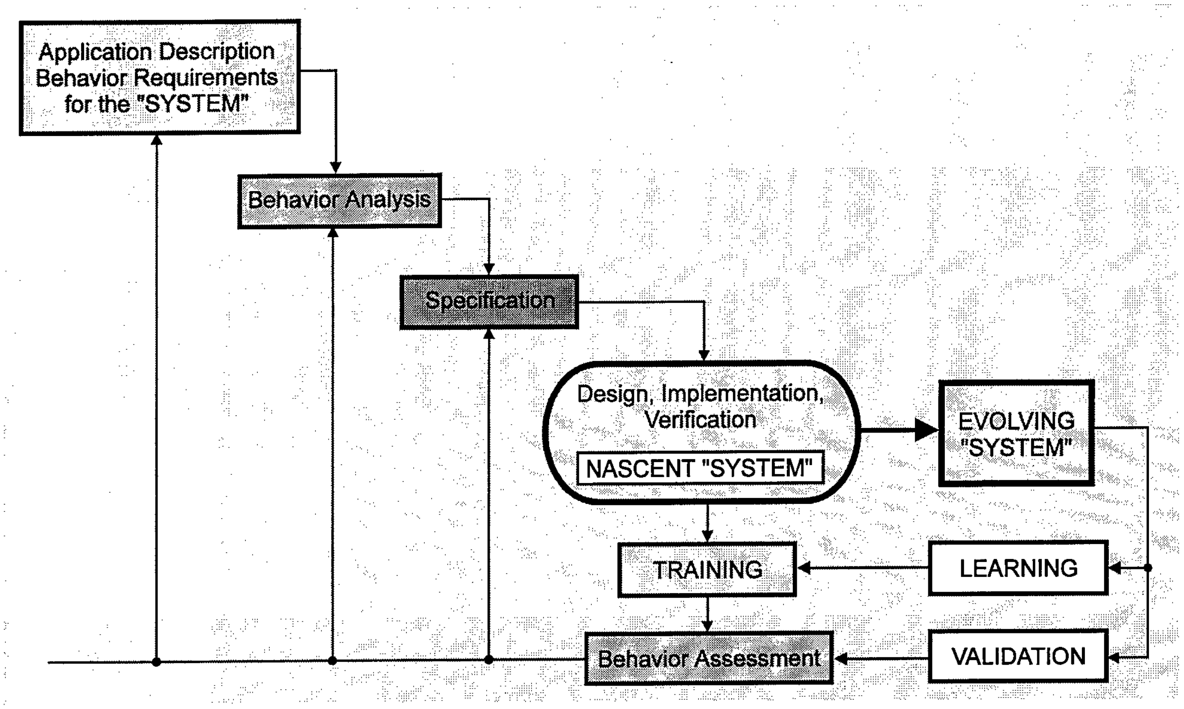


Figure 11: Global methodology in the engineering of the behavior oriented system

Based on a suitable training (learning) strategy the system acquires its knowledge during a training phase which is mandatory and prerequisite for appropriate behavior of the system. Training can usually be speeded up applying simulation including virtual reality. Within this context environments can be used that are much more changeable than the real ones.

After completion of training the behavior is assessed with respect to correctness (target behavior), robustness (target behavior vis-à-vis changing environment) and adaptiveness. Based on this assessment, further iterations during the engineering steps might become necessary in order to make the satisfactorily behaving system evolve from them in a step by step sense.

5 EMERGENCE OF AUTONOMOUS SYSTEMS

The computational techniques based on the new paradigm regarding brainlike processing structures, such as they have briefly been treated here, will significally contribute to the advent of future autonomous systems with revolutionary capabilities (Fig. 12, AUTARC: Autonomous artifical construct).

However, moreover, systems level research and development of critical experiments must be performed to fully examine the practical range of autonomous systems design issues. In this context validation techniques and certification methodologies associated with behavior and task-oriented autonomous systems are of particular importance.

Future research issues should include:

- Multi-Agent Systems,

- Autonomous planning,
- Adaptive multi-dimensional flight management techniques,
- Multi-Loop, non-linear, time verying global optimization techniques.
- Management of autonomy

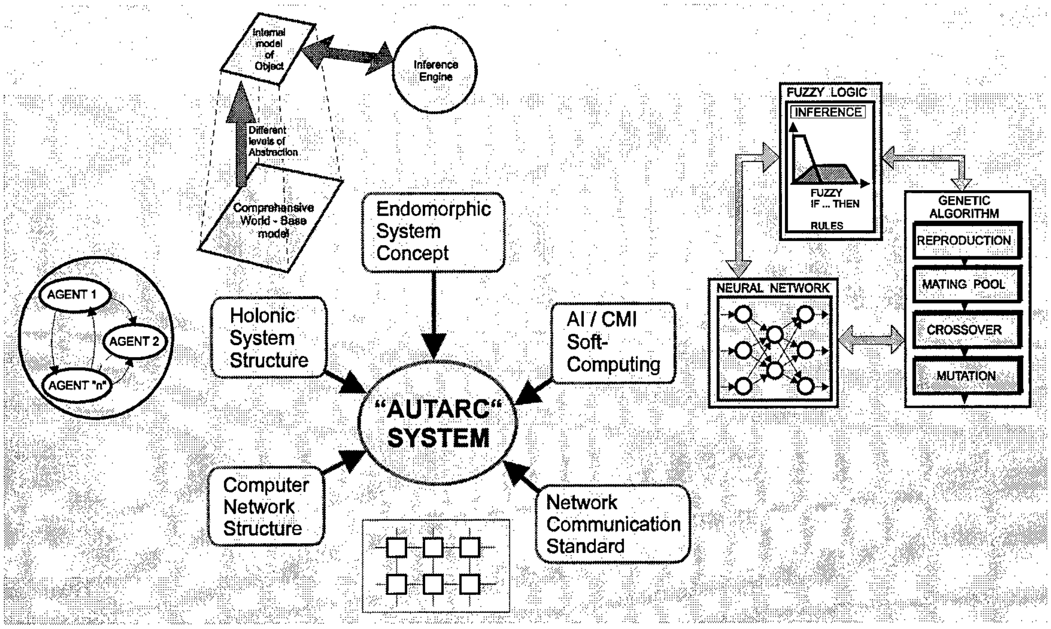


Figure 12: Enabling techniques for an AUTARC system

6 CONCLUSION

Central issue facing the NATO military community is AFFORDABILITY of aerospace systems.

Rapidly expanding technology base will enable fundamentally new system concepts, such as AUTONOMOUS UNMANNED COMBAT SYSTEMS (e.g. UTA) with high AFFORDABILITY.

There will be a PARADIGM SHIFT to BRAIN-LIKE information processing structures which offer a new quality of computational intelligence.

This yields self-learning, task- and behavior oriented control structures for autonomous systems that evolve, learn from experience and improve automatically in an uncertain, changing environment.

Literature:

[1] B.P. Zeigler et. al
Model-Based Architecture for High Autonomy Systems

S.G. Tzafestas (ed): Engineering Systems with Intelligence,
Kluwer Academic Publishers, Netherlands.

[2] B. Kosko
Neural Networks and Fuzzy Systems
Prentice Hall Inc., 1992

[3] U. Krogmann
Introduction to Neural Computing
AGARD LS179, Monterey, USA, Oct. 1991

[4] E. Sanchez
Genetic Algorithms and Soft Computing
1. European Congress on Fuzzy and Intelligent Technologies
7.-10. Sept. 1993, Aachen, Germany

[5] L. Steels, D. Mc Farland
Artificial Life and autonomous robots
Tutorial 11th. European Conference on Artificial Intelligence, Amsterdam, 1994

[6] U. Krogmann et. al.
Aerospace 2020, Vol. 3
Critical Enabling Technologies
AGARD, Paris, 1997

[7] U. Krogmann (ed)
Advances in Soft Computing Technologies and Application in Mission Systems
AGARD Lecture Series 210
Sept. 97, North York (CA), Amsterdam (NE), Madrid (SP), Ankara (TK)

[8] U. Krogmann
Towards autonomous unmanned systems
AGARD Lecture Series 210
Sept. 97, North York (CA), Amsterdam (NE), Madrid (SP), Ankara (TK)

[9] E. Westkämper et. Al
Holonic Manufacturing Systems
Symposium „Entwurf komplexer Automatisierungssysteme“
21.-23. May, 1997
Universität Braunschweig (GE)

Crew Assistance for Tactical Flight Missions in Simulator and Flight Trials

Dr. Axel Schulte and Dr. Wolfgang Klöckner
ESG Elektroniksystem- und Logistik-GmbH
Experimental Avionics Systems
P.O. Box 80 05 69
81605 München, Germany
E-mail: aschulte@esg-gmbh.de

1 SUMMARY

This paper describes an approach to technical crew assistance for tactical low-level flight missions. The relevant tasks are combined under the term *mission management.* In the first part of the contribution the chain of functions required for crew support in tactical mission management tasks is briefly summarised.

As the tactical mission management system is a major part of the Crew Assistant Military Aircraft (CAMA), its integration into the context of a cognitive assistant system is the subject of the second part of this paper. The modules of the tactical mission management system represent the methodological approaches and implemented functions of CAMA which are related to tactical operations.

A selection of pure tactical mission management functions and the fully integrated CAMA system recently underwent critical evaluation experimentation. Major parts of the tactical mission management system were tested in a flight trial campaign. CAMA was thoroughly evaluated in a series of simulator flights with operational personnel. The approach and relevant results for both activities are presented in the third part of this paper.

Finally, the conclusions and a view of future prospects are presented for tactical mission management assistance and the related research and technology programmes. CAMA will prove the maturity of its methods in forthcoming flight trials in which sensor integration issues will also be addressed.

2 INTRODUCTION

Pilots in tactical low-level flight missions in a multi-threat scenario under adverse weather conditions suffer a high workload due to the variety of tasks. Situations inevitably arise where the crew members are overtaxed regarding their limited mental information-processing resources, and thereby act erroneously. Because of this, situations may develop which jeopardise the mission success. The high crew workload is caused by the variety of different tasks, an insufficiently adapted crew interface, and the poor availability of information relevant to the situation and task in the cockpit. Closely related to this is the problem of situation awareness. Situation awareness has been achieved when the pilot has all relevant information which is required to resolve the most urgent task at his disposal [7]. This includes the objectively correct awareness, which is actually the most urgent task. Situation awareness is therefore the result of a continuous process of situation assessment [18].

Human-centred automation [1] appears to be a promising approach to the design of future cockpit avionics functions. The starting point is the analysis of the crew's tasks. Within the scope of tactical low-level flight missions the tasks can be summarised under the term *mission management.* These tasks comprise information gathering, situation interpretation and analysis including the ownship situation as well as tasks relating to in-flight mission planning and tactical low-level flight trajectory computation, and finally flight guidance and navigation tasks.

A technical system which is meant to be an effective crew assistant system for the tactical mission management should cover at least the above-mentioned chain of functions. Given this basis of crew assistance it ought to be taken into consideration that the assistant system should not replace the crew for certain tasks in terms of automation. The machine should be able to perform the tasks in parallel with the human operator and in co-operative function allocation [7]. Like the human operator, the machine part of such a man-machine system also needs to have the information relevant to the situation (task) at its disposal.

Therefore, two main issues have to be addressed in the design of an effective system: firstly, the functional aspects including the availability of suitable methods for information processing, and, secondly, the provision of comprehensive knowledge bases. In the field of tactical mission management these knowledge bases are closely related to digital terrain elevation and surface data.

The following section gives an overview of the required functional chain for an assistant system for tactical mission management, including some notes on terrain database integration.

The subsequent section deals with the integration of the tactical mission management functions into the broader view of the Crew Assistant Military Aircraft (CAMA).

Details of the various experimental evaluations, including the presentation of the results, are given in the next sections, following which the paper is summarised.

3 TACTICAL MISSION MANAGEMENT SYSTEM

This section will give a brief overview of the functional chain of a tactical mission management system for crew assistance. The functions were developed in the context of several projects concerning enhanced vision, crew assistance, night/adverse weather vision for transport aircraft and mis-

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

sion management. The underlying unified terrain database will be the subject of the second part of this section.

3.1 Functional Chain

The functional chain of crew assistance for tactical mission management reflects the tasks to be performed continuously by the crew in tactical low-level flight missions. The starting point is a valid flight plan, developed on the basis of the externally-given (by command & control) mission constraints. The mission constraints incorporate the specification of the targets or drop points, depending on the type of the mission. Furthermore, corridors for entering and leaving the hostile area as well as respective time constraints are given. In order to create an efficient plan in terms of survival probability an appropriate *assessment* of the threat *situation* in the operation area has to be taken into account, too. The *plan* can be *generated* either on ground or during flight. While performing the mission the pilot *interprets* the mission *plan* in order to derive the current tasks for flight guidance. Due to changes in the situation parameters, conflicts in the further plan execution are likely to occur. A typical example for such a conflict arises when a new threat pops up along the pre-planned track, and therefore further pursuit of the current plan seems unattractive. The crew's task in the context of mission management is to *detect* and identify the *conflict* and take all required steps to remedy the situation. In the example, an in-flight *re-planning* would be advised. The actual execution of the plan is a *navigational* and *flight guidance* task. A local optimisation of the flight trajectory might be desirable to ensure safe and successful mission completion.

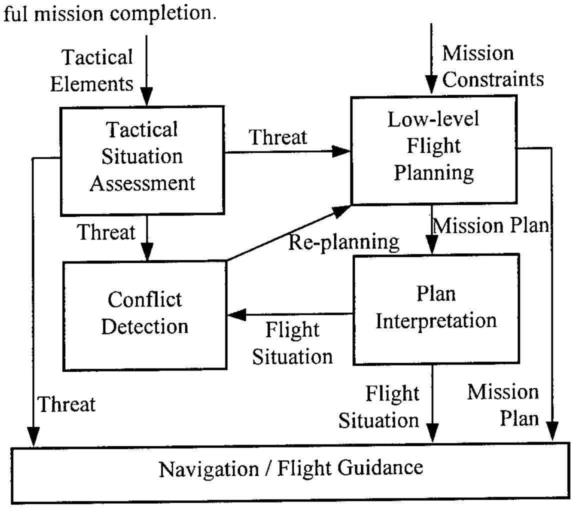


Figure 1: Tactical Mission Management Functions

Figure 1 shows the functional chain of the tactical mission management. The *Tactical Mission Management System* is designed as an on-board cockpit system to assist the crew, with a focus on the following tasks, as highlighted above:

- situation assessment,
- planning / re-planning,
- plan interpretation / conflict detection,
- navigation, and flight guidance.

The tasks will be performed by the crew and the machine in parallel. On the machine side the tasks are covered by the following functional modules of the Tactical Mission Management System [10][11]: The *Tactical Display* is the interactive navigation display serving various control purposes. The *Tactical Situation Interpreter* assists in the assessment of the external tactical situation resulting in a danger and threat analysis. The *Low-level Flight Planner* computes an optimal flight trajectory in terms of survival probability, according to the given mission constraints. The *Primary Flight Display* provides a three-dimensional visualisation of the terrain and advanced flight guidance symbology. The tasks of plan interpretation and conflict detection are not yet fully covered by the modules of the Tactical Mission Management System. These issues will, however, be addressed in CAMA.

The following subsections present some technical details about the four main functional modules.

3.1.1 Tactical Display

The Tactical Display provides the primary crew interface to the Tactical Mission Management System. Basically, it is an interactive electronic moving map display for navigational and operational purposes.

The module fuses the functionalities of the navigation display with interactive elements derived from a flight management system control and display unit by creating a bi-directional interface, so resolving the problem of the clumsy separation of flight plan manipulation and display in current cockpits.

The aim of the display-related aspects of the interface is the improvement of the pilot's situational awareness by depicting situation-relevant data. The actual information needed for preforming the task is highly influenced by the task itself. Obviously, the map information required for IFR flight is completely different from the information required in the context of tactical navigation. Typically, the differences are more subtle than in this example. The Tactical Display is primarily designed in order to cope with the advancing knowledge on human information processing [3], allowing the composition of the display contents from a vector oriented database. The display utilises digital terrain elevation data (DTED) and digital feature analysis data (DFAD) [19] in order to create a topographical map in any required scale and orientation. The various feature classes can be displayed selectively, providing very efficient de-cluttering of the screen contents. Military operations-related aspects are covered by the incorporation of tactical symbols. Further, the interpreted tactical situation is dynamically depicted utilising a three-dimensional threat coverage diagram. The threat map display can be activated for the different above-ground altitudes or be slaved to the aircraft's present altitude. The mission plan and an optimised ground track of the low-altitude flight plan are indicated.

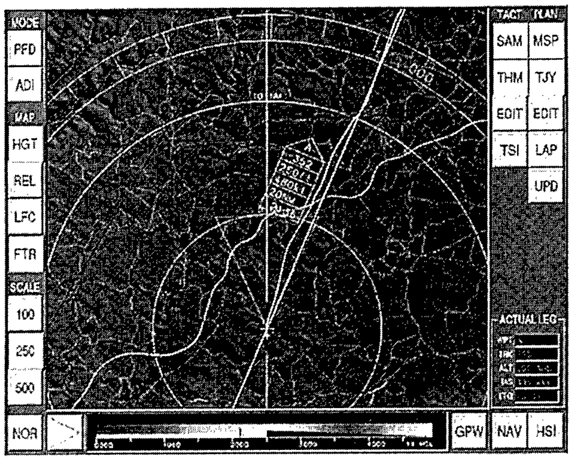


Figure 2: Tactical Display

Figure 2 shows the tactical display depicting the synthetic map. Various pushbuttons (operated via touch sensitive screen) implement the user interface.

In addition, the Tactical Display provides the control interface to:

- The Low-level Flight Planner. This allows the pilot to enter/edit waypoints interactively by marking them on the map;
- The Tactical Situation Interpreter, by supporting the manipulation of tactical elements.

Further details on these functions are given below.

3.1.2 Tactical Situation Interpreter

The Tactical Situation Interpreter is a knowledge-based module which forms part of the situation assessment function. Its main contribution is the computation of a threat map, which gives a penalty value distribution over the considered operation area.

Hostile or threatening tactical elements are passed to the Tactical Mission Management System via data link, or may be entered into the system through the Tactical Display. The tactical elements themselves provide little information to the crew in terms of decision aids. The tactical situation interpretation now assesses the tactical elements in the context of knowledge about the surrounding world, the threat's characteristics and the ownship capabilities and situation.

The calculations are based upon digital terrain elevation data and the threat models. Threats such as surface-to-air missiles are described by a set of typical parameters such as maximum range, operationability, variation of effectiveness with range, and models relating to threat area overlapping. Aeronautical constraints, restrictions, and tactical considerations can be incorporated as well.

Due to the characteristics of the threat's radar systems and the resulting radar shadows from the terrain structure, the altitude above ground up to which an aircraft remains undetectable by the hostile radar beams can be derived from the digital terrain elevation database.

The result of the tactical situation assessment provides much more task relevant information than the bare tactical elements overlayed to a map ever can. In this way, a valuable tool is provided for gaining a profound situation awareness regarding the tactical situation.

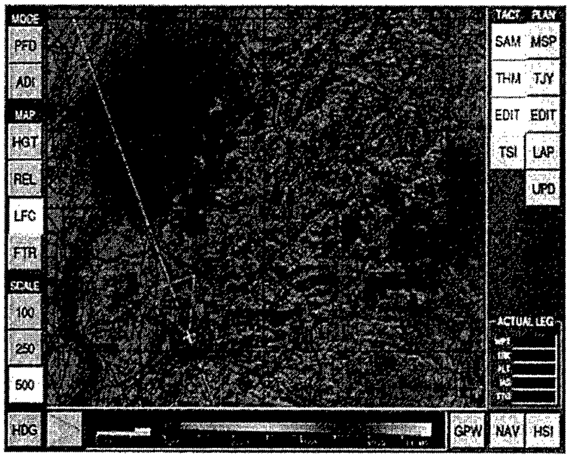


Figure 3: Controlling the Tactical Situation Interpreter

3.1.3 Low-level Flight Planner

The aim of the Low-level Flight Planner is the calculation of a three-dimensional route between the given mission waypoints with a maximum probability of survival in a hostile environment. This is achieved by avoiding threatened areas if possible, minimising the exposure to unknown threats and keeping clear of the terrain. Therefore, the mission constraints, the tactical elements and the resulting threat map, the terrain elevation data and the aircraft performance data are all taken into consideration.

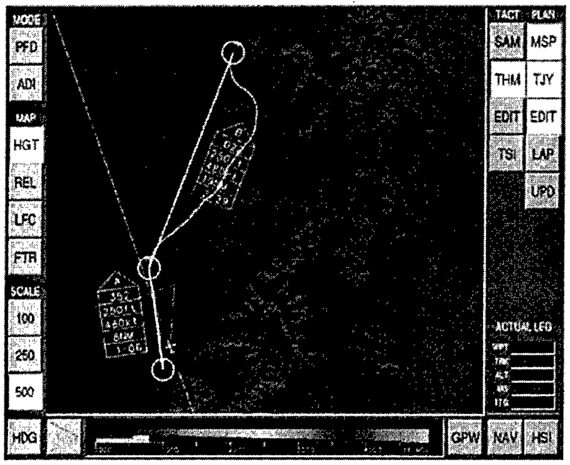


Figure 4: Controlling the Low-level Flight Planner

The system consists of three main functional sub-modules:

1. The *danger analysis* incorporates the threat map calculation as described before. Additionally, the visibility at each point is calculated without assuming any particular threats. The algorithm issues lower danger values on the side of valleys than in the centre. This behaviour reflects the pilot's low-level flying preferences. Finally, the danger analysis utilises the calculation of a ground collision probability, which is particularly high in rough terrain. This feature leads to generally higher flight altitudes in the absence of threats. An overall penalty value is calculated for each terrain grid point and stored as a danger model in an array.
2. The planner checks the flight status and assembles the target point and the planning area for the *optimisation* according to the mission constraints. The numerical optimisation is based on dynamic programming [6]. The optimisation provides an array of optimal directions to the target point. As long as a re-planning does not imply a new target point another optimisation run is not required. This means that the algorithm offers an optimal path from each point in the planning area to the desired target point. This characteristic of the algorithm provides one of the most powerful assistance functions of the low-level flight planner: the *rapid in-flight re-planning capability*. This function allows the generation of a new optimal trajectory starting at the aircraft's present position within a single second. In the case of intended or involuntary deviation from the pre-planned track, a recovery trajectory can be issued taking the terrain, tactical situation and mission goals into consideration. The function can be activated intentionally by the crew, by a ground proximity system alert or any other appropriate crew assistance function.

3. The *path selection* depends on the current planning mode (initial planning or re-planning). It constructs a terrain grid based flight path from a given start point or the present aircraft position to the target point. The *trajectory synthesis* assembles the low-level flight plan trajectory. In this way, a function is provided which assists on the skill-based human performance level [9]. During normal operation the pilot chooses a flight path by the consideration of relevant influences. The execution can be monitored by the crew assistant. In situations of increased workload it is possible that the pilot is no longer able to select a safe and efficient flight path. In this case the display of the automatically generated trajectory leads him safely to the next waypoint.

3.1.4 Primary Flight Display

Several scientific research studies start from the assumption that the pilot's information gathering from the out-the-window view is critical for flight guidance in low-level flight and landing tasks [3][16]. So, the incorporation of three-dimensional display formats into flight guidance displays is advised, in particular under restricted visual conditions. Classical flight guidance systems for instrument flight, such as a flight director display or an ILS indicator, provide minimal information, gathered by simple sensors from the real-world situation. This leads to low situational awareness. The pilot still has to learn adapted flying skills, instead of just utilising his natural and powerful skills of flying in a three-dimensional visual world.

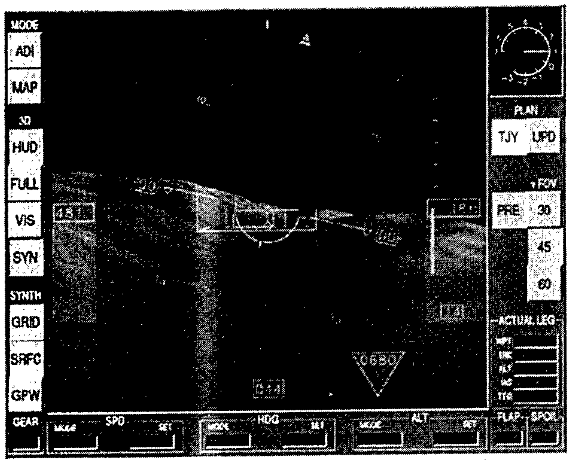


Figure 5: Primary Flight Display depicting terrain elevation

Figure 5 shows the Primary Flight Display with the three-dimensional graphical depiction of the colour-coded terrain elevation. The relief is enhanced by adding lighting.

The three-dimensional flight guidance display is a promising approach to the solution of the problems of poor visibility low-level flight. Its main element is the computer-generated three-dimensional cockpit view. It should be noted that the three-dimensional flight guidance display is not a visual simulation. It does not try to produce the most realistic representation of the out-the-window view, as in the visual systems of training flight simulators, but rather it produces a display carrying information relevant to the situation and tasks.

Figure 6 shows the most recent development concerning the synthetic three-dimensional vision displays for flight guidance. In addition to the pure terrain elevation depiction it incorporates the visualisation of the feature data and relevant surface models, showing the same scene as in Figure 5. The integration of terrain elevation and features for visual simulation purposes requires the merging of respective raw data streams in off-line procedures, as described in section 3.2.

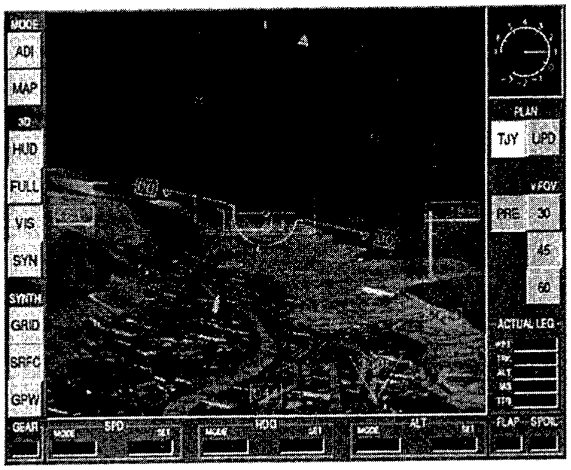


Figure 6: Primary Flight Display depicting terrain elevation, feature data, and textured surfaces

For most three-dimensional flight guidance applications the incorporation of sensory information is crucial. An *enhanced flight guidance vision system* comprises an imaging sensor (e.g. FLIR, mmWR, LL-TV) and the superimposition of the sensor image with the three-dimensional synthetic cockpit view. The incorporation of sensory data is essential. Due to incomplete or incorrect databases, the pilot cannot only rely on the synthetic image components. Inaccuracies of the navigation system yield another basic problem for pure synthetic vision systems.

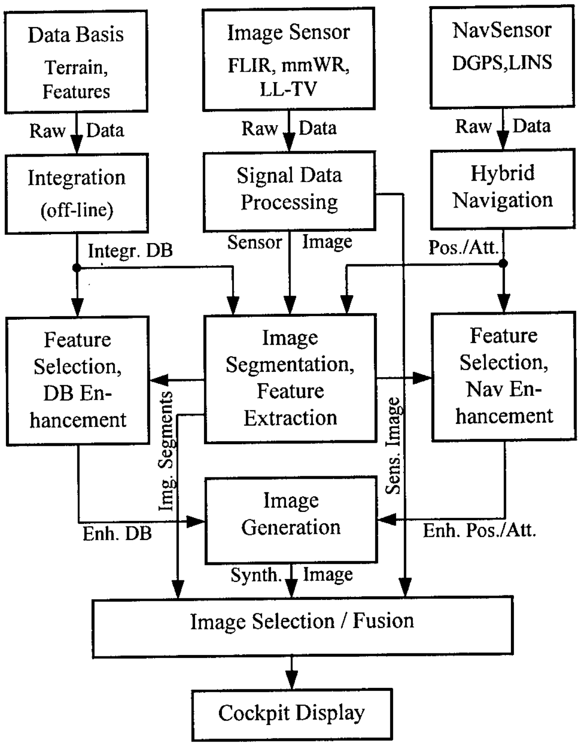


Figure 7: Enhanced vision concept

Figure 7 gives an overview of the components which might be suitable for an integrated enhanced vision system concept. In this concept the synthetic image generation is based on the use of an enhanced data base and an enhanced navigation system. The enhancements are conducted on the basis of sensor image segementation and sensor image feature extraction mechanisms [4]. Finally, the synthetic image, already containing the depiction of sensor image features, the sensor image itself, and relevant image segments are used in order to construct the cockpit display.

3.2 Terrain Database Integration

One major prerequisite in order to merge sensor image features with a terrain database, and to create a synthetic image as described above, is the availability of an integrated terrain database. In this context, the integrated database comprises digital terrain elevation data (DTED) and digital feature analysis data (DFAD) merged together, so that a geometrically unambigous description of the terrain surface emerges. The DFAD were processed first. In an initial step, all the area features were cut along the line features, ending up with smaller, non-overlapping area segments. In the next step the line features were converted into area features by giving them the correct width. In the following step the former line features were treated as the area features before, with the aim of removing all overlapping areas. The polygons thereby obtained underwent a triangulation process, as did all the uncovered areas in between. The elevation data grid structure was also taken into consideration. In a final step the whole two-dimensional surface description was elevated according to the terrain data, taking into account the relative elevations of e.g. dwellings and vegetation.

The computing of the integrated terrain database is fully automatic, from the raw data to the closed and unambigous representation of the three-dimensional terrain surface. It should be mentioned that the computing time for an appropriate gaming area is considerable, and the off-line calculation is mandatory.

4 CREW ASSISTANT MILITARY AIRCRAFT

The Crew Assistant Military Aircraft (CAMA) is a knowledge-based cognitive assistant system under development in close cooperation between the partners ESG, the University of the German Armed Forces, the German Aerospace Research Establishment (DLR), and DASA since 1995. CAMA and its philosophy have been described in various other papers such as [13][15], so that there is no need to go into detail too much.

CAMA incorporates the functional capabilities of autonomous situation assessment, including an individual model of the pilot's expected actions [14], machine recognition of the pilot's intent and errors [13], automatic tracking of progress in IFR and portions of the tactical mission plan, and detection of respective conflicts [15]. Detected conflicts are resolved by the automatic full mission planner [8]. Efficient information exchange between the crew and the assistant system is provided through a dialogue manager [5], taking the limitations of human information processing into account [2].

ESG contributes the functional modules to CAMA, which mainly perform the tasks related to tactical operations in the context of situation interpretation, planning, and crew interface, as well as the terrain database. Therefore, the modules Tactical Situation Interpreter, Low-level Flight Planner, and Primary Flight Display were derived from the Tactical Mission Management System and integrated into CAMA. The modules were enhanced by the addition of the following functional aspects in order to comply with the concepts of the cognitive assistant:

- The *Tactical Situation Interpreter* calculates the local threat distribution along the mission plan. Thereby, CAMA is able to perform the conflict detection with respect to local changes in the tactical situation.
- The *Low-level Flight Planner* calculates an IFR-plan-type abstraction of the low-level flight trajectory to allow CAMA to track the mission progress and derive the expected pilot's actions even in the tactical area. Furthermore, the Low-level Flight Planner allows the incorporation of corridors and tactical weapon delivery procedures, such as a payload drop procedure, directly into the flight trajectory. So, a homogenious unified flight guidance concept can be provided throughout the tactical mission.
- The *Primary Flight Display* was supplemented by an ADI display in order to support the IFR portions in flight guidance. Additionally, the expected behaviour parameters generated by the pilot model were displayed and respective deviations in the actual behaviour of the pilot were indicated. Finally, a terrain evasive maneuvre, generated by the terrain interpreter, will be integrated with the trajectory and displayed in case of ground proximity alert.

5 EXPERIMENTAL EVALUATIONS

Both the Tactical Mission Management System and the Crew Assistant Military Aircraft were evaluated by man-machine interaction experiments in 1997. The following two subsections present details of the experimental design and results are obtained.

5.1 Flight Trials for Tactical Mission Management

In order to evaluate the Tactical Mission Management System under real world conditions, flight trials were conducted in Summer 1997.

5.1.1 Apparatus

The experimental platform was a twin turboprop Dornier 128 aircraft provided by the Technical University of Braunschweig. It was equipped with a hybrid high precision differential GPS / laser INS navigation system. The experimental cockpit was equipped with a head-up mounted 13 inch high resolution LCD flat panel display. Figure 8 shows the experimental cockpit display obscuring the test pilot's out-the-window view. A Silicon Graphics Indigo2 High Impact graphics workstation hosting the experimental software was mounted in a shock-absorbing rack.

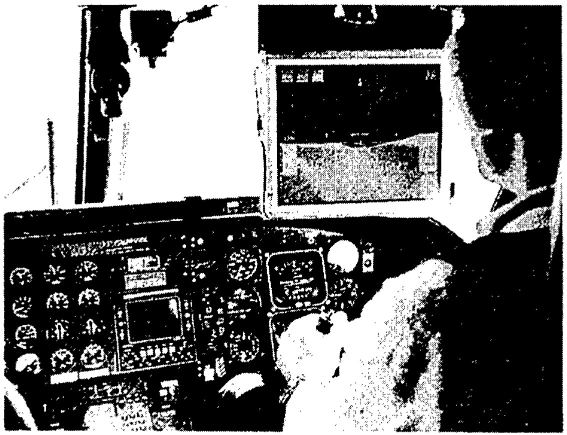


Figure 8: Experimental cockpit display

5.1.2 Subjects and scenario

The subjects were two scientific test pilots, one working as a safety pilot and the other as an experimental pilot. A total of six low-level flights were conducted at altitudes of approximately 500 ft AGL. The main task was to navigate in terrain proximity and perform terrain masking in mountain valleys. Point to point navigation utilising the visual identification of waypoints had to be performed. On-board planning and in-flight re-planning tasks were added. The mission contents were designed to take into account both familiarisation and training considerations and increasingly complicated tasks. For safety reasons the flights were performed under VMC. As they were meant to be under adverse weather conditions, the outside view of the pilot was obscured.

5.1.3 Evaluation results

Firstly, the flight trials were evaluated from the point of view of technical operability of the assistance functions under real-world conditions. This showed that it was possible to perform on-board full mission preparation and planning including autonomous trajectory generation. High precision visual navigation tasks with waypoint identification were successfully performed. The crew was able to cope with an in-flight change of the mission order by carrying out automatic full re-planning. Experimentally-induced deviations from the pre-planned trajectory could be recovered by the crew by use of the rapid in-flight replanning capability of the Low-level Flight Planner.

A second major aspect of the experimental plan was the system evaluation by subjective assessment, and questionnaires were completed by the pilots after the flight missions. In order to evaluate the situation awareness aspects the situation space was classified and structured into classes of situation elements. With regard to these classes the pilots had to indicate whether they had all the situational element-related information at their disposal whenever needed. Performance and benefits of functions were evaluated by listing all relevant tasks throughout the mission. The subjects had to comment on the quality of assistance offered by the Tactical Mission Management System with respect to these tasks. To evaluate the acceptance the pilots had to refer to a dedicated collection of the presented functions. The rating results are briefly summarised in the Tables below.

Situation Element	-2	-1	0	1	2
A/C Movement				*	
A/C Attitude (rel. Terrain)			*		
Mission / Task				*	

Table 1: Situational awareness

Task	-2	-1	0	1	2
Planning / Replanning				*	
Take off / Approach			*		
Transit Flight / Navigation				*	
Low-level Flight				*	

Table 2: Assistance Quality

Function	-2	-1	0	1	2
Low-altitude Flight Planner					*
Tactical Map / Mission Editor				*	
Interaction Concept				*	

Table 3: Acceptance

Generally, it can be summarised that the situation awareness achieved for aircraft attitude, altitude and speed was regarded as good. Concerning the ownship attitude and proximity with respect to obstacles and terrain, the synthetic vision format has to be improved. The depiction of the tasks and mission progress was regarded as very helpful. Generally an excellent overall pilot acceptance of the functions and formats was evident.

5.2 CAMA in Simulator Trials

After the demonstration of the technical feasibility of the selected mission management functions, a more scientific approach was chosen in order to evaluate the components and their effectiveness under a complex task scenario integrated in the cognitive assistant system CAMA. The simulator experiments were conducted in autumn 1997.

5.2.1 Apparatus

CAMA has been integrated and tested in the research flight simulator located at the University of the German Armed Forces, Munich. The dynamic model is derived from the ATTAS experimental aircraft of the DLR. A wide field of view visual simulation system is provided. The pilot's station is a generic glass cockpit equipped with monitors showing the Primary Flight Display, interactive navigation display, flight-log, and Radio Management Unit. Primary flight control is performed by a sidestick, an Airbus-type Flight Control Unit, throttle, etc. Information exchange between the pilot and CAMA is provided via touch sensitive screens and speech recognition / synthesis [5]. Figure 10 gives a good impression of the experimental cockpit.

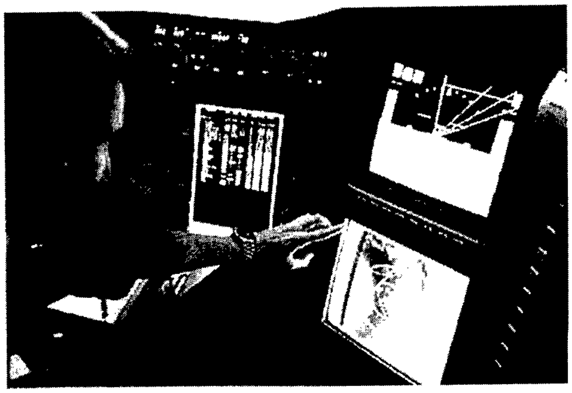


Figure 10: Research flight simulator cockpit for CAMA trials

5.2.2 Subjects and scenario

The subjects were ten German Air Force transport pilots (Airlifter Wing 61, Landsberg) between the age of 29 and 43 (mean age 38.4 years). The subjects were well experienced (combat ready) and had between 800 and 5300 total flying hours on the C-160 Transall (mean flying hours 2550).

The tasks were described by a full scale air transport mission, consisting of segments under IFR conditions followed by tactical low-level flight. The IFR scenario incorporated adverse weather conditions, high density airspace, varying availability of landing sites and extensive ATC communication. The tactical scenario was characterised as a dynamic multi-threat theatre due to the presence of surface-to-air missile sites, and tactical constraints such as air tasking order and airspace coordination orders. Regarding this, the experimental subjects had to conduct transit flight into the tactical operation area. After departure from a controlled airport the subjects had to perform several planning and re-planning tasks induced by thunderstorm areas and ATC requests. After passing the entrance corridor to the hostile area the mission continued in low-level flight avoiding known threats where possible, minimising the exposure to unknown threats and keeping clear of the terrain. Further, a tactical drop procedure had to be performed while being loaded by additional planning tasks concerning the low-level route ahead. Having left the exit corridor the pilot resumed the IFR flight. Converging traffic in terminal areas and closed airports forced the pilot to make various short-term and medium-term decisions.

Each subject had to perform the mission twice, once on each of the two experimental days. Time was given for familiarisation and training on the system.

5.2.3 Evaluation results

The evaluation results presented here focus specifically upon the subjective ratings given by the experimental pilots concerning the tactical mission management functions, including some general aspects. For information on a more general assessment of CAMA please refer to forthcoming publications from our group [12].

In order to assess the pilot's overall acceptance of the approach and the benefits offered by the system, a debriefing session concluded the experiments. First of all, it was necessary to ensure that the simulation environment was adequate for performing the required tasks. Table 4 lists the aspects of the simulation which are relevant to tactical low-level flight. The ratings were given within a range from –3 through +3, '+' indicates the mean result of the first trial and '#' stands for the second day flight with CAMA.

Simulation Element	0 1 2 3
Low-level flight / visual display	+ #
Landing / visual display	+#
Sidestick control	+#
Dynamic model	+ #

Table 4: Evaluation of flight simulation

In order to gain an overall impression of CAMA's performance, the following statements were evaluated. Table 5 lists the rating results, again measured on a scale from –3 (strong disagreement) to +3 (strong agreement).

Statement on CAMA	0 1 2 3
I always understood the actions of CAMA.	+ #
CAMA always seemed to understand my actions.	+ #
CAMA informed me well of my errors.	+ #

Table 5: Evaluation of CAMA philosophy

The methodological approach to the subjective evaluation of CAMA was very similar to the one practiced in the flight trials for the tactical mission management system. The scheme of the questionnaires has been derived from the following considerations. The questionnaires were structured according to the following aspects, as already mentioned in section 5.1.3:

- In order to evaluate the situational awareness aspects, the considered situation space was classified and structured into classes and sub-classes of relevant situational elements. Again, with regard to these classes the pilots had to indicate whether they had all the situational element-related information at their disposal whenever needed.
- Performance and the benefit provided by the functions were evaluated by listing all relevant tasks and sub-tasks throughout the mission. The subjects had to comment on the quality of assistance provided by CAMA with respect to these tasks.
- To evaluate the degree of acceptance, the pilots had to refer to a list of statements characterising the system behaviour and handling features.

The Tables 6 to 8 show the mean values of the ratings. It should be kept in mind that the Tables only depict the positive half of the ranking scale. Again, '+' indicates the mean result of the first day experiment and '#' stands for the second flight with CAMA.

Situational elements	0 1 2 3
A/C attitude	+ #
A/C position	+ #
Altitude	+ #
Speed	+ #
Terrain relief	+ #
A/C relative terrain relief	#+
Terrain / obstacle proximity	+ #
Missile sites	+ #
Threat efficiency in oper. area	# +
Momentary threat	#+
Threat at corridors	+ #
Threat along trajectory	+ #
Conflicts due to threat	+ #
Low-level flight plan	# +
Trajectory through terrain	+ #

Table 6: Evaluation of situation awareness

Task	0 1 2 3
Comply with mission constraints	#+
Identify waypoints	+ #
Assess threat efficiency	+#
Plan low-level flight route	+#
Perform ll flight guidance & nav.	+#
Re-plan trajectory to next waypt.	+ #
Identify own erroneous actions	+ #
Avoid erroneous actions	+ #

Table 7: Evaluation of assistance quality

Statement on CAMA	0 1 2 3
Behaviour was restrained	+ #
pleasant	# +
appropriate	# +
Handling was easy	+ #
pleasant	+#
not straining	#+

Table 8: Evaluation of acceptance

Finally, the displays were evaluated considering their contribution to situation awareness. The pilots had to give ratings concerning the depiction of certain display elements, again on a scale ranging from –3 (bad) to +3 (good). Table 9 shows the mean values of the ratings.

Display Element	0	1	2	3
Terrain			+	#
Trajectory			+	#
Waypoints			+ #	
Planned parameters			#+	
Autopilot modes			#+	
Autopilot settings			+	#
Threat map (Nav Display)			#+	

Table 9: Evaluation of display

A number of significant findings were derived from the Tables above:

1. Generally, a significant training effect was observed from the first day to the second day, the ratings improving noticably. It is significant that it does not only seem to be an effect of familiarity with the simulation facility and displays, but an apparent increase in understanding the aims followed by CAMA (see Table 5).
2. Concerning the Primary Flight Display, the results (see Table 9) show that after the short training period, the situation awareness ratings particularly for the spacially coded information (e.g. terrain relief) improved as opposed to the verbally coded [17] information (e.g. autopilot modes). To present three-dimensional information for flight guidance was a new concept for the subjects.
3. The situation awareness ratings concerning the tactical situation elements remained unchanged over the two-day experimentation at a very high level. On the other hand, the pilots situation awareness on primary flight parameters such as speed or altitude is certainly more crucial and has therefore been evaluated more critically.
4. The effectiveness of the tactical assistant functions gained constant high ratings.
5. The subjective sensation concerning the handling qualities of CAMA could be improved very significantly by a limited amount of training (see Table 8).
6. The overall somewhat hesitant acceptance shown in the results of the evaluation may indicate a certain amount of disapproval of new technologies on behalf of the pilots. It might also be the outcome of immaturity of the implementation of certain functions.

Generally, the results of the evaluation of the questionnaires and also of informal statements from the pilots can be regarded as very good. One experimental subject stated that CAMA gives him the opportunity to fly even more perfectly than he does anyway!

6 CONCLUSIONS

The results of the various experimental evaluations of the Tactical Mission Management System as well as the Crew Assistant Military Aircraft yield a wide variety of approaches and solutions in the field of crew assistance for tactical flight missions. The next two subsections conclude the paper by giving an overview and evaluation of the present work reported in this paper, and pointing out some future prospects for forthcoming developments.

6.1 Present work

In the present work the functions required in order to construct an assistant system for tactical mission management were derived from the tasks relating to tactical low-level flight missions. The result of this analysis shows that functions for situation assessment and mission planning make up the core of such an assistant system. Taking the requirements of human-centered automation into consideration, the cognitive assistant system CAMA provides well founded solutions. The integration of the tactical mission management functions into CAMA completes the system, so that an autonomous recognition of crew intent and errors as well as the detection of flight plan conflicts can trigger dedicated mechanisms of conflict resolution.

Selected functions of the Tactical Mission Management System were evaluated in flight trials. The results show that the on-board low-level trajectory planning and re-planning capability yields a powerful assistant function for navigational and flight guidance purposes and is well accepted by the pilots. In the flight trials the pure depiction of the terrain relief was used as synthetic three-dimensional flight guidance display. The comments of the pilots made clear that this format is not fully sufficient for low-level flight purposes.

In a subsequent experimental campaign CAMA and the relevant mission management functions were more throroughly evaluated in flight simulator trials. Ten professional military pilots performed complex missions in the simulator, assisted by the Crew Assistant Military Aircraft. In the de-briefing sessions the pilots gave extremely positive ratings on the system, its contribution to situation awareness, the quality of the assistant functions, and the degree of acceptance of such an electronic crew member. The effect of the minimum ammount of training on the system, in particular on the easyness of handling, is remarkable.

6.2 Future prospects

After a second experimental simulator evaluation in spring 1998, a flight trial campaign with CAMA is scheduled for early 2000. Further developments will focus upon the integration of imaging sensor information into the Primary Flight Display in order to achieve an enhanced vision system.

7 REFERENCES

[1] Charles E. Billings. *Human-Centered Aircraft Automation.* Technical memorandum 103885, NASA Ames Research Center, Moffett Field CA, 1991.

[2] Frank O. Flemisch and Reiner Onken. The Cognitive Assistant System and its Contribution to effective Man/Machine Interaction. In NATO System Concepts and Integration Panel Symposium. *The Application of Information Technology (Computer Science) in Mission Systems.* Monterey, CA, 1998.

[3] David C. Foyle, Mary K. Kaiser, and Walter W. Johnson. Visual Cues in Low-level Flight: Implications for Pilotage, Training, Simulation and Enhanced/Synthetic Vision Systems, In *48th Annual Forum of the American Helicopter Society*, Washington, DC, 1992.

[4] Simon Fürst, Stefan Werner, Dirk Dickmanns, and Ernst-Dieter Dickmanns. Machine Perception as Electronic Crewmember Capability. In: *The Human-Electronic Crew: The Right Stuff?* 4th Joint

GAF/RAF/USAF Workshop on Human-Computer Teamwork., Kreuth, Ge, 1997.

[5] Marc Gerlach and Reiner Onken. A Dialogue Manager as Interface for Communication between Aircraft Pilots and a Pilot Assistant System. In Proceedings of the 5th International Conference on *Human-Computer Interaction*, Orlando, FL, 1993.

[6] Ulrich Leuthäusser and Friedhelm Raupp. An efficient method for three-dimensional route planning with different strategies and constraints. In *Air Vehicle Mission Control and Management.* AGARD CP 504, Amsterdam, 1991.

[7] Reiner Onken. Basic Requirements Concerning Man-Machin Interaction in Combat Aircraft. In *Workshop on Human Factors / Future Combat Aircraft*, Ottobrunn, Ge, 1994.

[8] Thomas Prévôt and Reiner Onken. Knowledge-based Planning for Controlled Airspace Flight Operations as Part of a Cockpit Assistant. In *Air Vehicle Control and Management,* AGARD Conference Proceedings 504 of the 53rd GCP Symposium, Amsterdam, 1992.

[9] Jens Rasmussen. *Information Processing and Human-Machine Interaction. An Approach to Cognitive Engineering*, North-Holland, 1986.

[10] Axel Schulte and Wolfgang Klöckner. Perspectives of Crew Assistance in Military Aircraft through Visualizing, Planning and Decision Aiding Functions. In AGARD MSP, 6th Symposium on *Advanced Architectures for Aerospace Mission Systems,* CP-581, Istanbul, 1996.

[11] Axel Schulte. Cockpit Systems Design in Future Military Aircraft. In Proceedings of the 7th International Conference on *Human-Computer Interaction*, San Francisco, CA, 1997.

[12] Axel Schulte and Peter Stütz. Evaluation of the Crew Assistant Military Aircraft (CAMA) in Simulator Trials. In: NATO Research and Technology Agency, System Concepts and Integration Panel. Symposium on *Sensor Data Fusion and Integration of Human Element.* Quebec, Canada, 1998. In preparation.

[13] Michael Strohal and Reiner Onken. The Crew Assistant Military Aircraft (CAMA). In Proceedings of the 7th International Conference on *Human-Computer Interaction*, San Francisco, CA, 1997.

[14] Peter Stütz and Reiner Onken. Adaptive Pilot Modeling within Cockpit Crew Assistance. In Proceedings of the 7th International Conference on *Human-Computer Interaction*, San Francisco, CA, 1997.

[15] Anton Walsdorf et al. The Crew Assistant Military Aircraft (CAMA). In Proceedings of the 4th Joint GAF/RAF/USAF Workshop on *The Human-Electronic Crew: The Right Stuff?* Kreuth, Ge, 1997.

[16] Christopher D. Wickens, Ian Haskell, and Karen Harte. Ergonomic Design for Perspective Flight-Path Displays. In *IEEE Control Systems Magazine.* June 1989.

[17] Christopher D. Wickens. *Engineering Psychology and Human Performance*. Harper Collins Publishers, New York, 1992.

[18] Christopher D. Wickens. *Attention and Situation Awareness*. AGARD Lecture, University of the German Armed Forces, Munich, 1996.

[19] Data supplied by Defense Mapping Agency (DMA), St. Louis, MO, through Amt für militärisches Geowesen, 53879 Euskirchen, Germany.

Information, Decision or Action? — the Role of IT in Fast Jet Mission Systems

W. G. Semple
British Aerospace Military Aircraft and Aerostructures
Warton Aerodrome
Preston PR4 1AX
UK

Summary

Two distinct types of task can be identified in mission computing: providing information and automating action. Either may involve some form of decision making within the machine. The distinction between these tasks can sometimes be obscured in the search for a simple unitary architecture.

A common conceptual approach identifies a 'crew assisting' (CA) sub-system as a tool by which an operator, the pilot, operates on a workpiece, the rest of the aircraft and avionics. This conceptual disruption of the machine severely prejudices the design of the system as a whole.

There is not the clear boundary between information-gathering and decision-making that might be supposed from a purely functional inspection of the total mission task. On decomposing the IDA (Information, Decision, Action) cycle we find that an increased amount of 'smartness' in the preparation of information can reduce the quantity of 'intelligence' needed to make decisions.

Advances in software techniques and mathematical algorithms as well as increased computer power have enabled a richness of effective procedural computing that did not exist a decade ago, when contemporary capability suggested that future CA systems would be essentially AI.

By considering the information structures of the total mission task, and recognising the 'action' tasks as distinct and having subsidiary IDA cycles within them, most mission processing can be reduced to robust, deterministic, modular code. In normal operation, the remaining 'intelligent' processing requirement will be within the capacity of the crew, with less robust machine autonomy available for short periods of exceptional conditions.

1. INTRODUCTION

Many mission failures and aircraft losses have been attributed to low flexibility, high crew workload, lack of robust monitoring of the machine and its situation, or poor numerical support to resource planning or weapon deployment. In many cases, it is believed that a suitable computational aid would have averted or solved the problem, and it is probably safe to assume that continuing rapid advances in Information Technology will have increasing impact on the functionality, design and implementation of data processing and information systems in combat aircraft.

There is already great scope to apply the new technology to more or less conventional data management and algorithmic support. For example, a maturing generation of networking, data fusion and resource management algorithms are already being introduced in ground-based and multi-crew airborne applications. Their deployment in fast jets has been delayed only by the cost of upgrading the 'legacy' avionic systems currently widespread in this class of aircraft.

Other, more novel, applications lie in advanced functions of situational interpretation, mission and tactical planning and general support to the pilot, areas where closed mathematical functions can not be found or require undue computing effort, areas in which terms like 'artificial intelligence' and 'cognitive computing' are commonly heard. Here, functional concepts are not yet fully defined or agreed, and the feasibility — or at least the cost — and the practical use of many potential computer solutions are not fully understood.

Close parallels among many civil and military applications are enabling more or less rapid development and deployment of 'smart' or 'intelligent' IT, notably in applications such as diagnostics, vehicle management and communications, mostly in ground-based or multi-crew systems. But in mission systems for combat jets, the special factors of unique applications, an unusual and restrictive MMI environment, and very low product frequency have combined with reducing defence budgets to slow the maturation of the kind of integrated support system which would allow flexible, effective single-crew operation across the spectrum of missions.

Broadly, the technology which has been slow to mature can be categorised as 'Crew-Assisting' (CA) Systems. These are sometimes seen as modular components which will be added to the system in due course; such a view partitions the system into 'high-level' and 'low-level' computing functions as somehow fundamentally distinct, which in turn begs some design questions and constrains the design options.

This paper reviews approaches taken and the nature of the advances made, expected or still over the horizon, and considers desirable and achievable rôle models for IT in fast jets.

2. LEVELS OF COMPUTER SUPPORT

Before considering the rôle of IT in the cockpit, we will consider the skills and services that IT can bring to its rôle.

2.1. Computing Capability

As the term Information Technology has come to be used, computer processing — whether numerical, logical or associative — is at its core.

Traditional computing was essentially simple: numeric inputs were transformed to numeric outputs by clearly defined arithmetic processes. In practical systems, simple elemental processes would be aggregated into processes of great complexity; the underlying determinism, although always present, could be obscured by that complexity. For real time, multi-dimensional problems, applications were until recently limited to comparatively simple transformations.

Processor speed continues to grow, and more parallelism becomes available as the weight and cost of processors and memory reduce. Combined with novel algorithms, the increasing speed of succeeding generations of computer allows 'correct'

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

solutions to be found in 'real time' to an expanding range of complete problems where previously approximations had to be subjected to the operator's judgement. Examples are to found in network routeing, management of large fuel systems, weapon release, terrain screening and the like.

As we move up the 'intelligence' scale from the traditional 'dumb' processing we reach a 'smart' level where a process may still be strictly deterministic but has not been completely determined by its originator. Such systems include neural networks with fixed training sets, and rule-based systems in which not all combinations of interactions have been analysed. We can use such devices to solve complex problems where we can assess a solution but may not be able to write a direct procedure to find it. Examples of such systems are found in image recognition and system diagnostic applications.

Such systems are sometimes spoken of as simple forms of artificial intelligence (AI), but unless they are able to modify themselves after construction, they reduce to complex but 'dumb' systems which merely have not been fully analysed. Internally, they tend to draw on the work of the AI community for novel analysis and programming styles, but this does not, in most cases, alter their fundamental nature.

At the top of the scale is the Intelligent System. By 'intelligent' I mean able to appreciate situations and solve problems other than by following previously programmed instructions.†

Truly 'AI' systems could be flexible assistants to the crew, able, with appropriate training, to take on any *new* task. Such systems have yet to be fielded, although there are many systems operating in the 'smart' domain by mechanisms which it is hoped will lead to the 'intelligent' domain.

2.2. Styles of Usage

A distinction is usually drawn between 'data', taken to refer to low-level 'facts' in numeric, character string or boolean form, and 'information', which encompasses all facts including high-level statements of intent, or abstract or fuzzy characteristics such as 'A is usually more aggressive than B'.

Traditional data processing — the mathematical transformation of low-level data to low-level data — required only the first form. The computer could be used for a few fairly difficult calculations, for a lot of simple data management, storage and retrieval, or the output could be fed back into the control loop of an automatic system.

Now, techniques like data fusion, image processing and graphical interfaces allow machines fed on 'data' to produce quite high-level 'information' — for example, an assessment that 'targets are probably a formation of pattern X', or a graphical display of fuel trends.

Increasingly, we are able to generate 'information' using robust, mature procedures in which we can have full confidence. Sometimes, we must reach for 'smart' computing, notably for pattern recognition and matching or for diagnostic guidance, but even here systems are emerging in which we can place considerable confidence.

A difficulty arises with high-level control functions. As we seek to draw 'higher-level' — which in practice means summary or even judgementally interpreted — information from the machine, it becomes harder to ensure that all the inputs are correctly accounted and we tend to reach more for the 'smart' techniques in which our confidence is still not complete. We can draw information, even advice, from the system, but we become reluctant to give the machine the autonomous control which we are happy to allow its simpler counterparts.

2.3. Communication and Storage

Advances in data transmission and storage have been as important to airborne IT as developments in hardware and software have been, although they have perhaps received less attention.

The advent of the optical databus has raised the bandwidth of intra-vehicle communications a thousandfold. Taken with the capacity of multi-processor backplane clusters on which high-rate closely coupled processes can, when necessary, be co-located, we can for the present regard transfer capacity as effectively infinite for the majority of applications.

Inter-aircraft datanets are becoming routine, even among 'legacy' avionics. Inter-aircraft bandwidths are unlikely to approach those already possible within one vehicle, and constraints will continue from the need for data security and covertness, but we will certainly move towards more powerful multi-vehicle information networks than we have had hitherto.

The older aircraft in service today began their working lives with from, say, half a megabyte of semiconductor memory down to none at all. But about one day's average wage in NATO countries will now buy — retail! — a tiny card containing sixteen megabytes. Main memory is no longer much of a problem, and, with optical disc devices now sufficiently rugged for airborne use, nor is permanent bulk store.

3. MODELS OF THE CREW ASSISTANT

3.1. Background of Crew-Assisting Systems

Over time, a number of concepts have been formed of what a Crew-Assisting system would be. A strong, although not universal, tradition has grown up that, at least in those parts which interact with the crew, practical mission systems IT must be 'intelligent' in some sense, and should mirror the crew's perception of the world.

Iconic forms of CA systems have emerged from the DARPA *Pilots Associate* programme [1], the UK *Mission Management Aid* project [2] or French *Copilote Electronique* programme [3], and iconoclasts have emerged to challenge these. These programmes took pilot-centred, task-oriented or information-oriented requirements models in varying degrees, but, in line with the culture of their time, they and other contemporary programmes shared an AI foundation for the software.

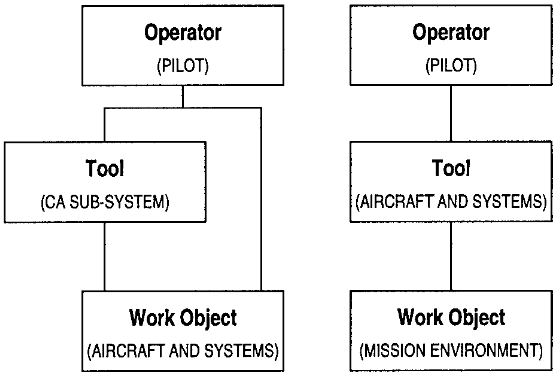
Subsequent developments in both conventional and novel hardware, in software technology (often deriving from AI work) and in procedural algorithms have opened up new possibilities for approaching the general requirement and it has become appropriate to review previous assumptions.

† There was once a fairly clear distinction between program and data from hardware up to problem definition. With the development of complex, recursive interpretive systems, associative memory and neural networks, etc, this distinction does not always remain above the hardware level. One could argue that this broadens the category of 'self-modifying' programs. For present purposes, we will continue to make the intuitive distinction between when self-modifying code is used merely as a way to simplify the tasks of design and implementation, and when it is used to create a 'free learning' system. Only the latter is taken as AI in the present discussion.

The term AI is sometimes applied to procedural systems which use bio-emulative techniques to implement the procedures, but any 'AI-ness' here is significant only to the AI academic, not to the end-user.

3.2. Hierarchical Models

We can compare two models of the position of a crew-assisting system in an overall hierarchy of operator, machine and environment. These are illustrated in figure 3.1.



(a) **"Crew-Assisting" Model**

(b) **Holistic System Model**

Fig 3.1. Approaches to Crew/Aircraft Interaction

In the first model, forms of which have been proposed by different authors, the aircraft is seen as a 'work object' on which the CA system is constructed to operate. The second model sees the aircraft and its systems as the tool by which the crew operate on the mission environment.

The two models need not be seen as mutually exclusive. Rather, they represent the extremes of a continuum extending from task-sharing between two crew members, one subordinate and electronic, to a hierarchical organisation with the human engaged only at the top. Nor must they *necessarily* lead to different implementations — the direct line from the pilot to the aircraft allows model (a) to be interpreted as a superset of model (b) — but in practice they support two distinct perceptions of the rôle of on-board computing.

By its nature, a tool is used to operate on an un-cooperative work object — an axe or a chisel is used to shape material which resists shaping whereas bare hands are used to shape clay, and computers are used for hard calculations but would not be employed to work out the product of four and five. The concept of a computer tool to operate the aircraft presumes that the aircraft is essentially inoperable, that it is a 'work object' rather than a tool in its own right.

If the aircraft is represented by generic current-day combat aircraft, then we are driven towards the CA model as a means to mitigate the difficulties that are encountered with the application systems. The issue then becomes: what is the functionality of the CA module?

We have a choice of rôle-model:

i) an 'intelligent' assistant;

ii) a computer toolkit, much as a suite of PC tools.

An approach to (i) would be to clone the crew in the CA, as nearly as technology allows, and for the human to employ the CA as he would employ a (less able) subordinate in an all-human organisation. The crew are able to by-pass the assistant when they see fit — as in most human organisations! However, the machine will not in reality approach the capability of the human in the domain of 'judgement' that is the human's particular expertise, nor will it have the diversity of skill and understanding. Human factors research consistently emphasises that *trust* is a key issue in the use of 'intelligent' systems. Further, if the machine is not sufficiently able, or if it is *believed* not to be sufficiently able, the crew will become patchily involved in details. When this happens, confusion soon arises as to who is responsible for what. This is to be expected from observing human organisations with similar mis-matches in ability or trust. Results of laboratory studies of depth-first and breadth-first automation structures carried out some years ago at British Aerospace have supported this expectation.

If we believe that option (i) is too ambitious in the short term, and that all machine functions related to the direct operation of the aircraft and weapons should be (broadly) deterministic, then the remaining option is the conventional computer tool.

Consensus has not been reached on whether a CA should be an assistant (a rôle which is consistent with the tool architecture) or a (set of) tools in the more conventional sense. This proved a topic of lively discussion in the final plenary session of the third Human-Electronic Crew Workshop [4]. If we adopt the 'toolset' model (b), the option opens to treat the aircraft and its systems as a whole, considering them as the tool by which the crew are to operate on their target and on the rest of the mission environment.

Once we have selected option (ii), then model (a) reduces to a disrupted form of the total machine concept (b) — although one that might be forced upon us if we wish to fit a CA capability to a legacy system. If we are considering a new system and if technology has not yet made the 'intelligent' option (i) available, we must do what we can with option (ii), and model (b) is then the logical choice. Even when robust 'intelligence' becomes available, there may be no reason to produce an 'intelligent' and therefore unpredictable system if a deterministic system is able to do the job.

By combining the approach to mission tasks discussed in 3.3 below with the new functional power of on-board IT illustrated in section 4, it appears that it is now possible to meet the needs of the mission with a system which is close to the integrated model 3.1 (b) — *in normal operation.*

There will be times when the human crew will become saturated or incapacitated, whether by demanding emergencies, unforeseen system malfunction, or enemy action.. Although we may have a mission set in which AI can not yet routinely replace the human crew, it is likely that it will soon be able to approximate human performance in most situations ... and certainly to exceed the performance of a disabled human. Thus we can envisage a secondary rôle for the IT as being a *spare* crew member, if not an assistant, much as the parachute is in a sense a 'spare' set of wings.

3.3. THE IDA CYCLE

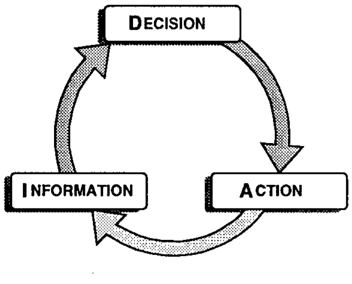


Fig 3.2. The IDA Cycle

Much attention has been given recently to the *Information, Decision, Action Cycle* or IDA Cycle model of military operations. The model proposes that all military activity consists of chained, nested and overlapping cycles of gathering information, making decisions on that information, and puting the decisions into action.

The model can be used to develop effectiveness studies by comparing the cycle times of opposing forces. For present purposes, however, we will consider in more depth the structure of the model rather than competitive cycle times.

We can decompose the activity in each of the three main phases of the cycle.

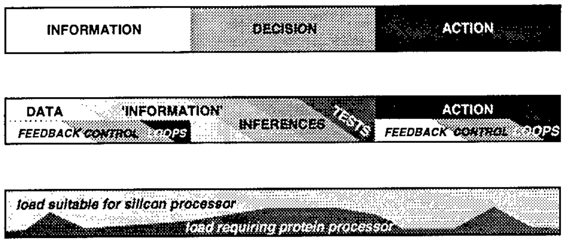


Fig 3.3. IDA Decomposed

Information

At the start of each cycle comes getting the information — intelligence gathering. Usually, acquiring information requires one or more subsidiary IDA cycles. We must know what information is needed, we must decide how to get it and then we must do the getting.

Sometimes, the information wanted is 'what if' rather than 'what'; this will usually call for analysis or modelling. While IT does not yet offer much support to logical analysis, it is well able to provide any associated numerical support and has long been a primary resource for modelling.

Decision

Some decisions present no difficulty, as for example *if traffic light turns red then stop* (usually).

Other decisions are more complex, but what might appear to be a complex decision, requiring much time and effort, often turns out to be a quantity of information processing with a simple decision at the end. We receive 'raw' data which does not support a decision we have to make and recast it, analyse it, transform it and mix it until it supports the decision. Graphical presentation can often be invaluable, because our brains have remarkable ability to see trends 'at a glance' which may not be immediately apparent in the original data. Fuel states which would otherwise require intensive calculation can be anticipated from graphical displays of planned usage through the mission with overlays of actual usage to a current point. If the machine can do the necessary processing and presentation, the decision-making itself may require little time or effort. Indeed, it may turn out that the machine can do that too.

There are decisions that perhaps only the human can make. Developing the traffic-light example: *if traffic light turns red and there is a 38-ton lorry eleven feet behind and the road is greasy and I'm driving on my employer's insurance and there is a police car sitting at the junction and I already have 10 points on my licence then ... ?* The decision requires not only calculation of the risks of collision by stopping or not stopping and the risk of subsequent injury but also the disparate factors of the employer bearing the cost of damage and the probability of legal consequences to the individual. On the other hand, from the employer's point of view, perhaps this is a decision that an objective machine should make!

The extreme case is when a decision can not be expressed as the selection of one among many pre-determined options, but requires the creation of an innovative plan. If tasks and scenarios are trained, rehearsed and practiced, such decisions are comparatively rare, but when they occur the extra effort is largely applied to answering the 'what if' questions discussed under *information* above.

Action

Actions range from the very simple, such as pulling a trigger, to the complex and prolonged, such as constructing a new air base. The more complex an action is the more it decomposes into subsidiary IDA cycles. Landing an aircraft is a continuous loop of acquiring procedural and dynamic information, deciding on change to speed or attitude, corrective actions (some of which may be reflex) and actuation of the controls.

Review

We find a recursive structure in IDA in which each element is built of all the elements, with the *receipt* and *processing* of data or information being substantially the largest constituent. IT systems are good at that. Many decisions are entirely mechanistic and can safely be entrusted to the machine. So the computer can do these too. If the information is appropriately formed and presented, many other decisions become self-evident or 'intuitive' and thus involve acceptable workloads and timescales. Decisions involving trends — fuel usage against plan, progress towards energy advantage — can be assisted enormously by graphical presentation, another IT strength which has only recently become feasible to any degree in fast jets' cockpits. Much of 'action' has always been automated; IDA analysis can guide us in simplifying the management of the automation without either sacrificing determinism or unnecessarily taxing the operator.

4. TECHNICAL ADVANCES

This chapter considers examples of recent advances in what is achievable in airborne IT. It is impossible to give an exhaustive list in the space available, but these examples should serve to demonstrate how far our ability to provide the crew with a manageable interface to the 'mission machine' has advanced in the last decade or so.

4.1. Enabling Technologies

Before considering specific mission functions it is perhaps appropriate to illustrate the developments that are taking place in useful technologies intermediate between generic aspects of computing and function-specific techniques and algorithms.

Neural network techniques have been progressed and simple networks are finding their way into modern sensor processors, but such techniques are not usually sufficiently analysed or robust to dislodge more conventional computing from the bulk of mission processing. But an interesting departure from mainstream neural network philosophies, and something of a breakthrough in pattern searching, has come from the AURA (Advanced Uncertain Reasoning Architectures) project at York University [5].

The AURA project tackled both underlying mathematics and *low cost* hardware design to develop a rapid pattern-searching and matching engine for use with incomplete and uncertain data. Building on earlier work on associative memories it owes more to database theory than to mainstream neural network concepts. AURA applies *binary* networks to a novel extension of pattern encoding and superposition techniques to combine partial and combinatorial matching with great search speed. For example, the prototype from the first phase of the project verifies 4000 addresses by checking for *inexact* matches (to locate *eg* typing errors) in under 10 seconds, several orders of magnitude faster than alternative methods.

Apart from speed, AURA has two key features which make it attractive for mission systems. The first is that it handles incomplete or uncertain data as well as it handles perfect data. It will find all available matches to the data patterns given, and it can be asked to report matches to any *m* or more of *n* available data — thus not all *n* need be correct for a search to succeed. The second is that, although a statistical error rate is introduced by the encoding and superposition technique, all errors are 'false alarms' — no valid matches are missed. Since incomplete and uncertain data will in any case require more rigorous inspection than simple pattern-matching, this error form is consistent with the problem itself and is not a disadvantage in the matching engine; AURA will quickly reduce large candidate pattern sets to small ones, robustly filtering the input to the weightier algorithms used for the main analysis.

AURA is currently able to handle symbolic and numeric data and can be extended to handle image data. Its potential applications range from pattern recognition in images (and other large sensor datasets) to identifying rule-oriented tactical characteristics from geometric and low-level identity data.

4.2. Data Fusion

A great deal of effort has been applied to data fusion over many decades and many techniques have now matured for plot, track and identity fusion for single and networked aircraft. Techniques continue to mature towards a whole-battle integrated capability.

The outgoing generation of data fusion was constrained (by the low computing power available) to employ simple gating techniques for association, and even the tracking filters could not be too ambitious, for example with regard to multiple behaviour hypotheses. Fusion of images, with other images or with vector data, was all but impractical. Data tended to be fused mainly within a single sensor; the nature of the equipment and the federated systems in which it was installed limited (although did not entirely prevent) sensor-to-sensor fusion.

With the processing power and main memory now available we are able to install capable and advanced data fusion with statistically sound association algorithms and increasingly flexible and robust tracking algorithms. More traditional sensor data systems, requiring significant operator involvement, remain in service only because the rate at which capability has grown has outstripped that at which older systems are updated or replaced. Pointers to the data fusion capabilities of new aircraft such as EF2000 and F-22 are widespread in the literature.

Image fusion has even higher processing demands than more conventional track and identity fusion and current prototypes are still mainly confined to specialised ground machines, but progress here is rapid indeed and many capabilities should be serviceable within five years. Functions include fusion of map data and sensed images to obtain vector overlays, either to enhance the image itself or for precise navigation, and identification of target formations from incomplete or noisy sensor returns (in both optical and RF domains) [6, 7, 8]. Novel enabling technologies such as AURA can make a significant contribution to this kind of function.

As the problems with single-platform fusion were solved, attention naturally turned to multi-platform distributed systems where the fusion task is spread among co-operators. An operational requirement is that their pictures of the world must remain consistent with each other. There is also an economic and logistic requirement that such a system should as far as possible be 'plug and play'. That is, where the specific characteristics of a particular sensor must be modelled that modelling should if possible be confined to the processing in (or at least associated with) that sensor — it must not permeate the data fusion system as a whole.

Again, new algorithms to 'automate' the data system are becoming available. Notable here is the work on distributed data fusion by Durrant-Whyte at Oxford and the associated work at British Aerospace's Sowerby Research Centre [9].

4.3. Tactical Situation Assessment

Tactical Situation Assessment is a large topic, but some examples drawn from air-to-air and air-to-ground missions should convey the extend to which computer assistance can now be made available to the combat pilot.

4.3.1. Defining the Situation

It is often valuable to predict whether there is a line of sight between two moving points, such as co-operating aircraft, or to know the heights of the terrain shadows from the position of a given observer (such as a tracking radar) within some specified radius of the observer.

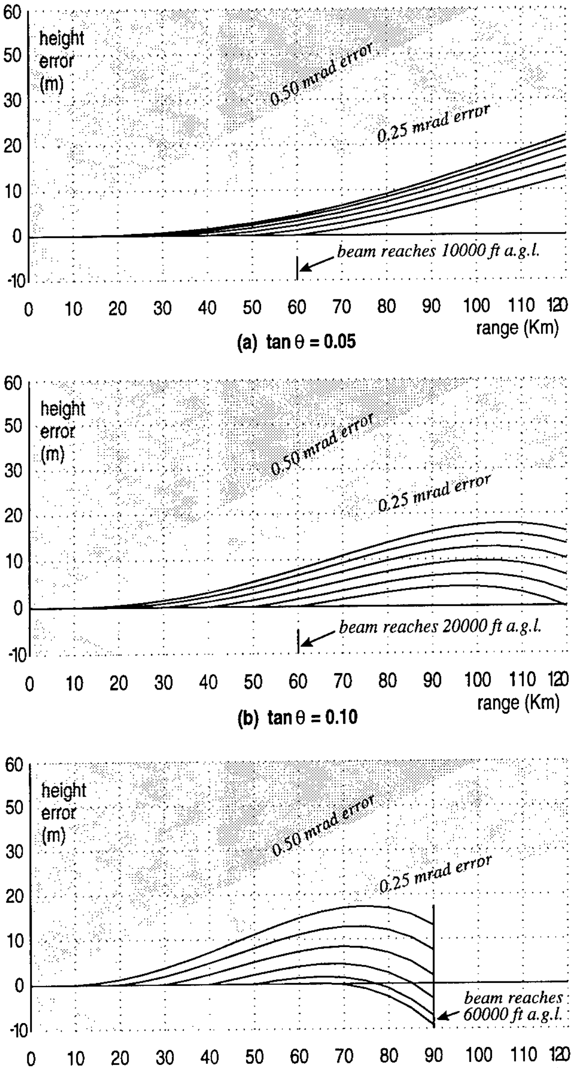
Codes for this 'intervisibility' problem have been around for a long time in ground installations, but rapid-response airborne functions have been impossible or impractical because of the amount of data storage, data transmission and processing that is needed. Storage and transmission are no longer sources of difficulty, and nor is processing speed. Various proprietary codes exist with different performances, and most are not published, but an intervisibility code written recently at British Aerospace is given as an example.

To compute terrain shadow heights from a given observer position, generating a circular terrain mask, requires less than one microsecond per grid point on a 133 MHz Pentium PC — not exactly a top-of-the-range machine nowadays. To generate a display mask of 25 Km radius at 100-metre grid postings requires $250^2\pi \times 0.85\,\mu s$, or one sixth of a second. The time required in a future modular avionic card will be very much less. The shadow heights thus calculated can be used to produce 'what-if' displays for any enquiry height, as well as for the nominal path height, by simple thresholding against a variable demand height in the graphics module.

The calculation includes second-order integration, allows for the differences in surface curvature north-south and east-west, treats beam slope exactly and has a good approximation to the beam refraction in an assumed exponential atmospheric density profile. Ridge diffraction is not included.

The accuracy of the algorithm, assuming an exponential atmosphere, is illustrated in figures 4.1.

The graphs show the errors in the shadow heights algorithm for initial beam gradients of 5%, 10% and 20%, with shadows generated by obstacles (ridges etc) at ranges of 10Km to 60Km from the observer. At the 20% beam slope the calcula-



(a) tan θ = 0.05

(b) tan θ = 0.10

(c) tan θ = 0.20

Fig 4.1. Errors in fast beam/shadow propagation code

tions have been truncated at a height of 60,000 feet (18,000 metres) above the observer, reached at about 90Km range, because by this height the exponential atmospheric datum against which we are are assessing the algorithm has become meaningless, and so has the whole concept of terrain screening! It will be noticed that these results go a considerable distance outside the range/height envelope that will usually be required of such an algorithm, and that in typical applications, say up to 50 Km and a few thousand feet of height, the calculated heights are in error by only one or two metres. Errors due to uncertainty in the air density distribution at the required time and place will usually exceed these, so there would be no point in any more exact calculation.

4.3.2. Interpreting the Situation

It is important in air combat to maintain a perception of where and when own ship will be able to launch a successful attack against its targets, and of where and when it will itself be vulnerable to attack. Graphical presentation of missile launch success zones can enhance the pilot's appreciation of the situation, although a display of where the zones are 'now' leaves much for the brain to do to build the picture it needs of how the engagement might evolve.

The computer can offer a presentation which in many cases comes closer to the information that is actually required, by transforming the tactical data in time before presenting it.

The principle can be illustrated by a simple example. Suppose we are in a boat on one side of the English Channel, which we wish to cross on a roughly diagonal course. Our desired course crosses busy longitudinal shipping lanes and also a number of transverse ferry paths. We wish to maintain a distance of at least R from all other vessels (R can vary according to the class of vessel).

Assuming we have a capable sensor suite with full data fusion and tracking, we can construct a computer plan-display showing all 'targets' and tracks, and each target can have around it a disc indicating is 'exclusion zone'. We will see a chaos of moving discs. We will be able to 'see' a short time ahead, but it will be difficult to see whether a course will remain outside all the discs as we progress.

Now suppose that instead of displaying simple radius discs around each target, we show zones which highlight points which, if we were to proceed directly to them at our intended speed, would be within R of another vessel *when we got there.* (These new zones will not of course be circular.) So long as the other vessels maintain course and speed, the zones will change only slowly as we move, and, critically, on whatever track we are moving, *the intersection of the zones with that track do not change.* If a 'target' changes velocity, then its zone will of course change, but that change will be conspicuous among the slowly changing generality of zones.

That concept forms the basis of the *TACMAP* visual aid developed at British Aerospace. The 'zones' in *TACMAP* are lethality and vulnerability zones (green and red respectively) for pilot-selected missile types against a small number of primary targets. The zones can be calculated for maximum range, 'no escape' or whatever intermediate criterion is required. A typical plan display is illustrated in figure 4.2.

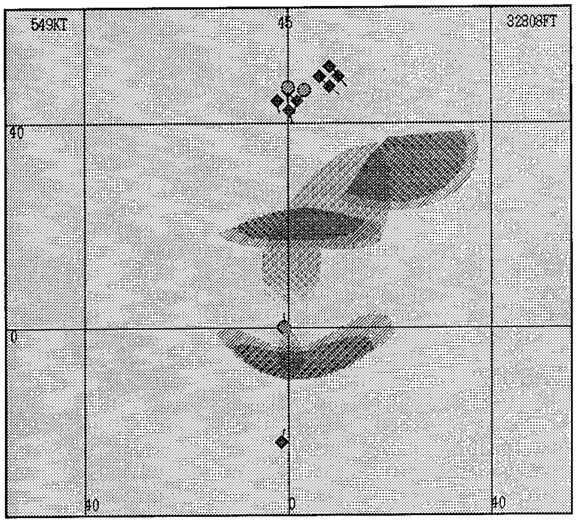


Fig 4.2. *TACMAP* LSZ Display

To date, simulator trials have been conducted with two-dimensional displays (plan or inclined plane, and / or vertical section at steerable azimuth). Recent developments in the algorithms, together with increases in processor performance, now make the calculations for three-dimensional displays supportable when a 3D format is required.

These time-projected displays are not of course a complete solution — there will be uncertainties in how the targets will behave which invalidate the assumptions in the machine, and the crew will also want to make their own judgements from the present situation. However, a display such as *TACMAP* can take out a large part of the cognitive workload where there is any degree of stability in the target behaviour, and equally where the evolving situation is to be assessed for any assumed target behaviour.

Once again, the machine's contribution is strictly procedural: the 'rules' by which it transforms the data are simple and understandable, although the arithmetic to calculate the missile envelopes can be lengthy.

4.3.3. Monitoring the Pilot

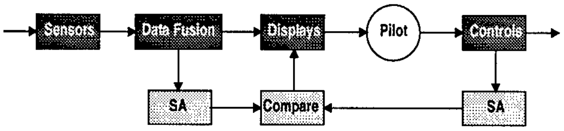


Fig 4.3. Illustrative Crew Monitor

So far, we have discussed how the on-board IT can help the crew to assimilate the mission environment and plan how to deal with it. It is to be expected that with a good, manageable machine, the crew will make few tactical and procedural errors. But human errors will still happen, and the machine may be able to alert the crew in time for recovery to be made.

Various types of 'pilot error' have been identified over the years. Let us assume that those attributed to unfriendly equipment or to poor automation will reduce in the normal course of design evolution, and that lack of information will be addressed by better information systems. Simple forgetfulness is already addressed in scheduling and 'reminder' cues in aerospace as in other domains. Saturation in extreme situations will be rare if the IT system is doing its job; when saturation occurs, despite the best efforts of a deterministic system it may be that further help will be needed from something a bit cleverer than just more of the deterministic system.

This leaves the common problems of task and target fixation, in which important incoming data are overlooked and important but 'routine' actions are neglected. The same symptom can occur at times of high workload when the pilot is not strictly saturated to the extent that he requires an 'intelligent' aid. An example would be for a pilot concentrating on his primary target to overlook the rapid promotion in the threat table of a hostile who was previously thought to be un-threatening.

Figure 4.3 illustrates one model by which the machine could monitor the crew's behaviour. The machine knows what information has been presented, and it know what actions the crew have carried out. If the machine can detect nothing in the crew's actions to indicate that some possibly important information has been registered, it can try a little harder to draw attention to that information. Perhaps the size or brightness of a symbol or of some text could be increased, or a flashing arrow could point to it. *In extremis* a major stimulus — visual, audible or even electromechanical! — could be delivered, but to do this the machine would have to be *very* sure of its ground. A noticeable change in the background to one's concentration is usually acceptable, but an unjustified interruption is not.

There is a clear distinction between this function, which seeks only to make information more noticeable, and a definite interruption as from a missile approach warner or a fire alarm. The essence is that with modern displays we have the means to make *mild* attention-getters from which a modest false alarm rate is more acceptable than from traditional strong attention-getters.

4.4. Tactical Decision Aids

Tactical Decision Aids have been a fruitful source of controversy over the years, and more work must to be done before the requirements are fully resolved.

Advisory or Informative ?

While we may seek to capture the best in theory and practice and to train aircrew accordingly, engagements can evolve other than in 'the book', and aircrew vary in their tactical skill.

If some crew are less skillful in tactics than others, it follows that there is a rôle for a coach, advisor or whatever — but what kind of advisor? Trials on both sides of the Atlantic have produced the following results. First, inexperienced crew make more use of tactical advice from a computer than experienced crew who, in the main, *do not like* advice from the machine. Second, their dislike has been justified, because (so far) they can usually do the job better!

What crew both ask for and require is the right *information*, when they need it, in a quickly and intuitively assimilated form. Current aircraft would be hard pushed to provide this in close combat — human eyes are more effective than current sensors here — but beyond visual range and some way into visual range, there is a lot that the machine can offer.

Recent work reported by DERA [10] has confirmed and developed earlier hypotheses about how an air combat aid might operate. A tactical computer program called *JOE* was written to provide more players in multi-crew air combat simulations. With development, *JOE* became fairly proficient, but could not beat the best human pilots. The team went on to analyse and quantify different aspects of *JOE*'s performance, and found that its weakness was only in respect of tactics selection; *JOE* out-performed all the human pilots when it came to execution of the tactics.

Thus suitable tasking for the IT system in air combat might be that illustrated by a suggested development of *COMTAC*, an early research programme on cockpit *COM*puter *TAC*tics at British Aerospace [11].

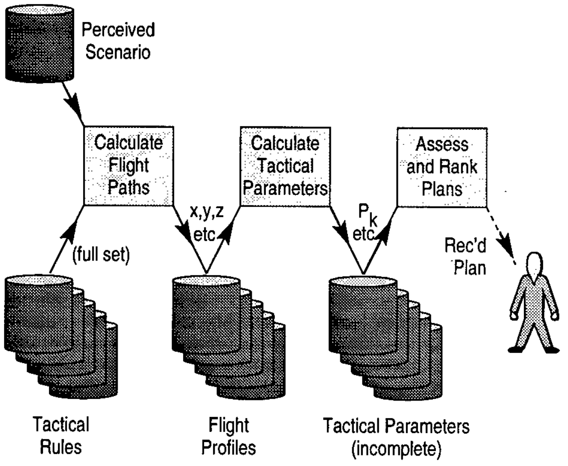


Fig 4.4. The Original *COMTAC* concept

Figure 4.4 illustrates the original, *advice* or *decision*-based concept. The machine would compute tactical parameters (kill probabilities, fuel usage, targets of opportunity, exposure to threat, *etc.*) for a range of tactical possibilities and recommend the tactic which achieved the highest 'score'. (The diagram is somewhat abstracted — the number of tactical options would in practice be pruned continuously as the calculations progressed.)

For the pilot to accept the advice given, he must be reasonably confident that all relevant parameters have been correctly taken into account by the computer program. With advice in this form, he has to accept or reject it in full — and as has been found in many research programmes, a fully trained pilot is unlikely to accept it.

Now consider the *information*-based variant illustrated in figure 4.5.

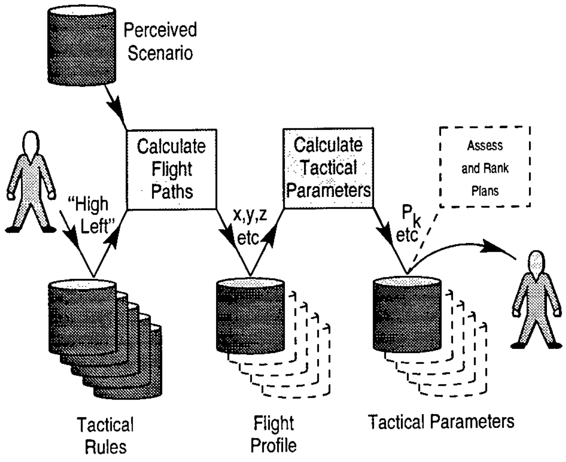


Fig 4.5. The Revised *COMTAC* concept

Here, a change has been made at the start of the chain. The pilot enquires about a specific tactic. The machine then calculates and assesses various tactical parameters. (If required, it can do a more complete job, since there are now fewer flight paths to be computed, but we may wish to take a faster response time rather than to enrich the calculations.) Finally, a crucial change is made at the end of the chain. The selection operation is omitted entirely, and the output is no longer "this is what to do" but "here are the results of the calculations", which might be a flight profile to maintain radar cover, a statement of fuel required, speed and altitude at missile launch, time of missile flight, or whatever.

Many of the results can be presented in graphical form for at-a-glance interpretation. The essential difference lies in the fact that they are now specific procedural results, obtained by clearly definable arithmetic processes. The 'trust' issue has been overcome, because the machine is operating in its natural mode. The pilot can make coarse mental adjustments and/or get the machine to re-calculate for a modified course.

4.5. Mission Plan Maintenance

Attention has been given recently to the possibility of reducing the elapsed time to accomplish a mission by carrying out all or part of the planning process while the task force is airborne, by transmitting a ground-computed plan to the aircraft or computing it on board.

Once the plan is established, by whatever means, some maintenance will often be required during the course of the mission. The scheduled time over the target might change, hostile action might force an unplanned diversion from which it is required to regain the planned route, or a deliberate diversion might be made round newly discovered threats. In all these cases, waypoints, speeds and fuel plans are liable to adjustment even if the coarse plan is generally retained.

For a number of years 'flyable' computing has been able to support the crew in tasks like waypoint editing and in the resulting time and fuel calculations; deployment has been slowed by available MMI devices and legacy data architectures more than by what is now quite a modest processing requirement. But direct calculation of optimum flight paths has been seen as more futuristic. Work has progressed worldwide in establishments, industry and academia to explore and develop diverse methods derived from control theory, network engineering, heuristic and other AI, neural networks, *etc*, with confidence that expanding computer power would eventually make such calculations feasible in the cockpit.

Not only has the power of the machine become greater but, as in other applications, the demands of the algorithms have become less. For simpler route - optimisation problems it is now possible to obtain results very quickly on even modest machines. The following examples illustrate current developments of the minimum-cost routeing method proposed in [12].

Suppose we have an isotropic cost function $\mu(x, y, \ldots)$ such that the cost of any path F from A to B is given by

$$cost_{AB} = \int_F \mu(x, y, \ldots)\, \mathrm{d}s$$

where ds is the incremental distance along the path. The cost function can represent a mixture of exposure to risk, fuel rate, distance from desired track, weather adversity, *etc*. It can be shown that if F is cost-stationary (a necessary although insufficient condition for F to be the global minimum-cost path) then the curvature of F at any point P on F is given by

$$\underline{\kappa}_P = (\mathrm{I} - \underline{\hat{V}}\,\underline{\hat{V}}^T)\,\underline{\nabla} \ln \mu$$

where $\underline{\hat{V}}$ is the local direction of the path. Other functions can be derived which constrain the behaviour of more complex cost distributions.

This constraint on the path geometry can be used to construct an algorithm in which the number of speculative branches in the search process varies with the square of the distance rather than with distance as a power. The effect is dramatic. Figure 4.6 illustrates a route calculated with a cost function obtained by adding terrain height to a background value chosen to impose a reasonable penalty on distance covered. Thus the route minimises a weighted mixture of distance covered and the height of the terrain crossed. The route was calculated in a little less than one fifth of a second on a 133MHz Pentium PC.

In figure 4.7 values representing the threat from a tracking radar were added to the the cost distribution used in figure 4.6. The cost profile of the threat is roughly dome-shaped, falling towards maximum missile range, and has been filtered by a terrain shadow mask calculated as discussed in section 4.3.1. The colours are lost in the greyscale reproduction, but an increasing red-ness, which can be inferred from the darkness level, represents the cost density field while the contour lines in the figure represent the terrain altitude only.

The algorithm adjusts the step sizes automatically throughout the search to ensure stability and adequate granularity. With

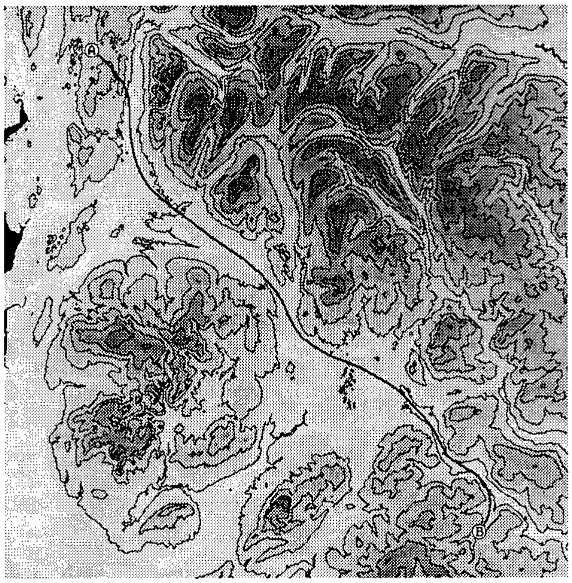


Fig 4.6. Route from a simple low-ground algorithm

the steeper cost gradients and finer-grained feature in the figure 4.7, the path was produced with 195 defining points whereas that in figure 4.6 was produced with only 150. The number of search steps increased more rapidly as the algorithm probed the threat distribution for weakness, and the execution time rose sharply to a little under half a second. Some experimentation with conventional search heuristics and the rules setting step length indicates that this execution time can be roughly halved during further development.

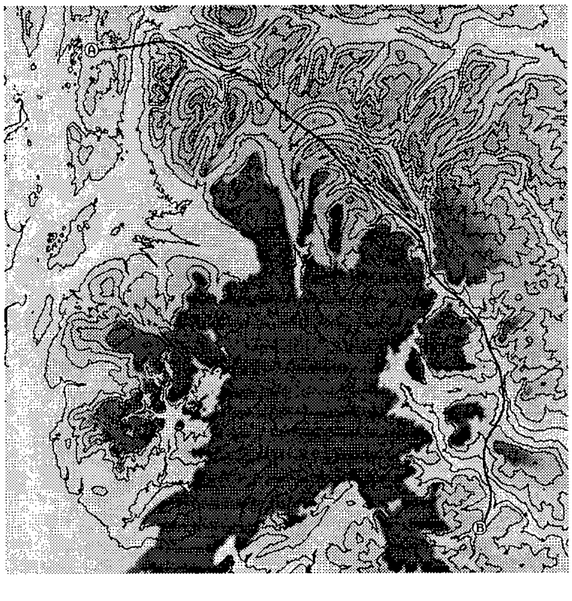


Fig 4.7. Route with SAM threat from same algorithm

Once a minimum-cost route to a given destination has been calculated, intermediate results have been calculated through the search space from which other routes starting some distance from the first can be generated with very little further calculation.

Suppose that the crew decide to continue on the original route *AB* from figure 4.6 after the threat appears, but want a continuous update on the computed avoidance path. Figure 4.8 shows a family of routes calculated from points along the original path. For clarity, the terrain contours have been omitted from the picture. No work has been done yet to obtain an *efficient* algorithm for these derived paths yet the average time to produce them is only 0.007 seconds. So while any cost field remains constant, we can comfortably maintain a continuous revision of the optimum path from current present position to the next waypoint — even within the refresh rate of the display screen, if required!

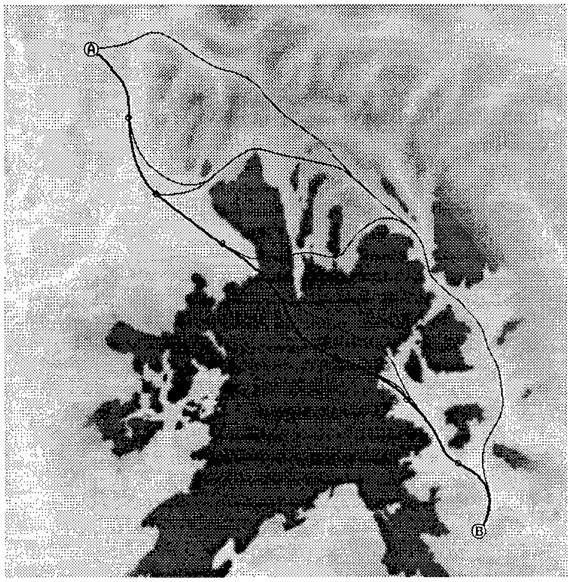


Fig 4.8. Route updates along fixed track

4.6. Display Management

We are able to generate well-processed information describing the mission situation and the vehicle status; we are able to produce predictive results for further queries that the crew might raise; we are able to make some assessment about whether the crew are operating within their capacity and whether they have made any obvious omissions. We are able to take on most of the routine aspects of vehicle and sensor automation (reliable avoidance of un-briefed low-level obstacles being probably the only major gap remaining) but the crew will of course require control and status feedback of the automation process.

All this amounts to a great deal of information available for display and there just isn't a lot of space in a fighter cockpit to display it. Here lies one of the last unsolved problems. There is no reason at present to believe that it either cannot or will not be solved, although solved it must be if single-crew aircraft are to be operated to greatest effect from deterministic data.

Normanton has pointed out the danger in an 'Electronic Co-pilot', or EC, forming an information layer between the crew and the avionics [13]. With an integrated procedural machine, the same danger may be distilled into the display manager — indeed, it was said during the time of the MMA project that "the pilot interface manager is the MMA", although a counter from an MMI specialist was on the lines that "with a properly integrated system there would be no need for an MMA".

To some extent, the machine will have to decide what is displayed. But the dangers here need not be so great as might be supposed; generic guidelines on what is displayed can be quite prescriptive in practice, so the crew can retain close and intuitively manageable control of the display content. Where the machine may have to be given more (apparent) autonomy is in the precise formatting, both where information should go and how it is rendered (adjusting size, contrast, etc).

The management of such issues within the machine is seen as an important link between Human Factors / MMI concepts, which are being widely researched, and the display needs of individual functions. The changes that have taken place in the last decade mean that old assumptions must be re-visited here, just as with the underlying crew-assisting models of figure 3.1. The issue is at the core of a recently started collaborative programme between Smith Industries and British Aerospace [14].

It is possible that reliance on deterministic data may itself depend on 'AI' management of the displays, but the evidence from progress in other functions suggests strongly that this will not be the case. With advances in display devices, application graphics, software design and engineering methods, and the growing (although still not universal) acceptance of iteration as a method for both requirement and design [15], it is likely that a mechanistic approach to the machine and its information can be carried through to the display of that information.

5. CONCLUSION

Mission Operations can be reduced to acquiring information, deciding on the correct courses of action, and carrying out these actions. *Decisions* reduce to (a) re-forming the information and (b) either trivial 'test and branch' procedures or complex judgemental assessments. *Information Gathering* and *Actions* usually decompose into subsidiary IDA cycles of revising information (feedback), revising decisions, and lower-level actions which, at the last stage of decomposition, emerge as activation or adjustment of mechanical effectors.

Advances in IT have brought a dramatic increase in the proportion of the necessary mission information which can be generated in the machine. Simple procedural decision-making has always been within the recognised competence of computers, as of simpler artefacts before them (examples here include alarm clocks, safety valves and thermostats).

By matching current or imminent IT capability to the requirements emerging from decomposed IDA, we will be able to produce airborne weapon systems which are manageable in themselves, with the characteristic desirable in all tools that their actions are known and understood, rather than having to build intermediate tools of indeterminate complexity to help manage inherently unmanageable aircraft.

There will always be circumstances, especially in combat, in which the pilot will lose his or her grasp of the situation. For such occasions, technology can offer a machine able to take on the conduct of the mission, at least for a short period. It will not be as competent as a human controller operating a near-perfect information system, but it will offer a fallback capability much advanced on an unaided and incapacitated human.

References

1. *Pilot's Associate: Evolution of a Functional Prototype*
C. S. Lizza, S. B. Banks and M. A. Whelan,
AGARD CP-499, Paper 16, May 1991

2. *Modelling the Information Flow — Development of a Mission Management Aid for Future Offensive Aircraft*
J. M. Davies, Proceedings of the 3rd International Workshop on the Human-Electronic Crew, Cambridge, UK, Sept 1994, DRA/CHS/HS3/TR95001/01, Jan 1995

3. *Copilote Electronique Project*
G. Champigneux and T. Joubert,
Proceedings of 4th International Workshop on the Human-Electronic Crew, Kreuth, Germany, September 1997.

4. Proceedings of the 3rd International Workshop on the Human-Electronic Crew, Cambridge, England, Sept 1994, DRA / CHS / HS3 / TR95001 / 01, January 1995

5. *Advanced Architectures to Support Intelligent Command and Control*
J. Austin,
Final Report on EPSRC Grant GR/K41090, 1997

6. *Sensor Motion Refinement through Viewpoint Correction and Robust Interpretation of Normal Flow*
D. R. Parker and J. P. Oakley,
IEE conf pub 443, Vol. 1, 1977

8. *Using Neural Networks as a Part of a System to Recognise Formations of Aircraft*
P. R. Zanelli and J. Austin, IEE pub 440, 1997

9. *Decentralised Algorithms for Tracking and Data Fusion*
H. Durrant-Whyte and P. Greenway,
Royal Aeronautical Society conference *The Rôle of Intelligent Systems in Defence,* Oxford, March 1995.

10. *Development of a Tactical Advisor for Air Combat*
R. D. Harrhy and R. Nield,
Proceedings of 4th International Workshop on the Human-Electronic Crew, Kreuth, Germany, September 1997.

11. *Computer Aided Tactics in the Cockpit*
N. Mitchell, AGARD CP 440, Paper 25, October 1988

12. *Optimum Routeing - Analytical Constraint of Search Space*
W. G. Semple, AGARD CP 563, Paper 10, January 1995

13. *Information Attributes and the Electronic Crewmember*
T. H. Normanton,
Proceedings of 4th International Workshop on the Human-Electronic Crew, Kreuth, Germany, September 1997.

14. *Aspects of the Crew Interface for Mission Systems*
C. R. Ovenden, W. G. Semple, K. M. Wykes, T. H. Normanton,
Agard CP-600, Vol. 1, Paper A14, Dec 1997.

15. *Cockpit Usability — A Design Checklist*
J. Turner, Agard CP-600, Vol. 1, Paper A22, Dec 1997.

KNOWLEDGE BASED DECISION SUPPORT TDPs FOR MARITIME AIR MISSION SYSTEMS.

H. Howells
Lt A. Davies RN
Systems Integration Department, Air Systems Sector
Defence Evaluation & Research Agency, Farnborough
Hampshire GU14 OLX UK

B. Macauley
Directorate of Future Systems (Air) 2
MOD(PE) Bristol BS12 7DU UK

R. Zancanato
Cambridge Consultants Limited
Cambridge CD4 4DW UK

ABSTRACT

This paper describes the range and scope of laboratory prototype Knowledge Based Decision Support systems for Maritime Air applications conducted by the KBS Group at DERA Farnborough UK. The rationale behind the choice and development of the requisite tools and software are described. The applications include Decision Support for Anti Submarine Warfare (ASW), and Anti Surface Warfare (ASuW) together with an ASW/ASuW Technology Demonstrator. Also included are the design and early development of a Knowledge-Based Decision Support System (DSS) laboratory demonstrator to support near littoral helicopter based Airborne Early Warning (AEW) operations. The scope of the application is large, encompassing: Detection, Situation Assessment, Reporting, Tactical Decisions, Fighter control, Radar/sensor handling and Airmanship. The design has focused on the requirement to provide extensive modularity to support extensibility and component reuse (both of the underlying expert knowledge and the associated implemented modules) in the proposed follow on TDP.

1.0 INTRODUCTION

1.1 KBS Group Background

The Knowledge Based Systems Group within Systems Integration Department at the Defence Evaluation and Research Agency (DERA) in Farnborough have been engaged for more than a decade in research and applications of knowledge based decision support. (Ref 1) The initial impetus was the general search for a means of managing aircrew workload. Mission systems were being specified for airborne use where manufacturers claims for improvement in mission performance were high but without corresponding assessment of the role of the aircrew required to attain such performance levels. From earlier work by the core team members on less sophisticated airborne systems the indications were that the newer proposed mission systems would generate a wake of different, additional attentional demands. The emergent technology of expert systems was then considered as a potentially useful avenue to explore.

1.2 Shells and Limitations

The then new LISP based Expert Systems shells such as Inference Art and Intelicorps KEE, together with lesser known proprietary products were acquired and experience gained in diagnostic level problems such as replicating aircraft warning panels. Experience in devising and using structured interview techniques in problem identification and assessing performance in Human Factors studies where more reliable measures were not available enabled the team to build skeletal laboratory demonstrators when coupled with the commercially available shells. Whilst the early experiences with the shells indicated the potential of the approach in functionally representing the required salient features in narrowly focused airborne domains the software speed limitations rapidly became apparent.

1.3 Application Expansion

It was realised that the fundamental design of LISP posed an impediment to the real time demands of airborne applications. To extend the laboratory demonstrators to capability which would interest military customers would require faster software and prototype build methods far beyond those commercially available. The group proposed that more systematic methods of knowledge acquisition, a

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

more appropriate form of validation methodology were needed for Knowledge Based Systems (KBS) development together with a real time capability more in line with airborne requirements. This would require considerable research investment, but coincidentally, the United States Government General Accounting Office (GOA) report published in 1981 drew attention to financial implications of continually assuming that the increasing significant elements that system designers were unable to specify in complex military systems could be compensated for by the ingenuity of human operators.

1.4 Realisation

Following the publication of the GAO report, a NATO Working Group was set up and reported in 1984 that KBS should be examined as one possible means of addressing such problems. (Ref 2) One of the authors was a member of the NATO Working Group and used the rationale for the need for research funding that if expertise was needed to generate complex systems, if that expertise could be incorporated in computer code within the mission systems as advice then overall mission performance ought to improve. Such arguments were successful and paved the way for the large scale UK-MOD funded research programme covering KA methods (PC PACK) (Ref 3) real time software (MUSE and D-MUSE) (Ref 4) and a validation methodology (VORTEX) (Ref 5) which currently are being applied to demonstrators for knowledge based decision support for Maritime Air applications.

2.0 THE FIRST MARITIME AIR APPLICATION - ANTI SUBMARINE WARFARE (ASW)

Previous background by the author in human factors assessments of Maritime Air sensor and mission systems led to an awareness that the task was characterised by a developing situation where fine judgements were examined rather than the more deterministic reactions encountered in strike mission management. When an application was needed to evaluate the capability of the validation methodology research which recommended a spiral development life cycle for rapid prototypes for KBS workstation demonstrators then the developing nature of the maritime mission was seen as an appropriate application to demonstrate the concept of KBS. (Ref 6) The skeletal Anti- Submarine Warfare (ASW) scenario used proved doubly effective in demonstrating the tools designed for validating the knowledge base and when combined with the measures of effectiveness defined by the maritime customer illustrated that the tactical advice offered exceeded the customer expectation of the laboratory demonstrator. (Ref 7) The Validation of Real Time Expert Systems (VORTEX) application had been written in LISP due to the groups experience with LISP based shells. At that stage it would have been difficult to make the case for funding a separate line of software development due to the large scale US-DARPA investment in LISP during the initial phase of the Pilots Associate Programme. (Ref 8) However, the Group's experience in sponsoring the development of the real time software development toolkit (MUSE) and its successful application in a laboratory demonstration of a multi engine helicopter warning panel and its performance on tapes provided by NASA Ames on telemetered systems status data from the X29 research aircraft provided sufficient evidence in its potential to secure additional funding. The next expanded version of the ASW application which focused on producing a decision support system for controlling more than one platform used MUSE rather than LISP.

3.0 THE SECOND MARITIME AIR APPLICATION - ANTI SURFACE WARFARE (ASUW)

At the same time that the group was developing the LISP funded ASW demonstrator for the validation methodology evaluation, parallel effort was also being expended in applying MUSE to a skeletal mission manager workstation demonstrator using a fixed wing strike scenario. (Ref 9) Compared with earlier success of using MUSE with diagnostic applications with multi engine helicopter warning panel and the NASA X29 data the different data types and varying input frequencies began to reveal limitation in the real time performance of the MUSE software. Additional research funding was then received to develop a multi agent real time capability (D-MUSE) to maintain the software performance against the more demanding scenarios envisaged. Maritime air experience during the Gulf War had revealed the difficulties in tracking high speed fast patrol boats which would also camouflage their presence by mooring alongside oil rigs or inserting themselves in slow moving fishing fleets. This application was considered ideal to assess the real time capabilities of the multi agent software and so a knowledge based Anti Surface Warfare (ASuW) decision support system workstation demonstrator was built. (Ref 10)

4.0 AIRCREW ROLES IN ASW/ASUW AIRCRAFT

Maritime aircraft are required to operate world-wide, often at short notice and in a variety of roles. To fulfil such demanding requirements the aircraft carry very sophisticated sensors and complex systems which are configured and managed by the aircrew to most effectively meet the needs of the varied missions. Such variation in operating environments and improved capabilities of future mission systems led to the increasingly demanding role for aircrew. This role is characterised by:

1 A need to adequately consider the most appropriate mode in which to operate a sensor due to the increased number of modes.

2 The need to handle a vastly increased volume of data provided by improvements in sensor performance. This is generated by the increased number of targets likely to be seen over longer ranges. The lower signal to noise ratios at which detection is possible also increases the number of false contacts.

3 Greater uncertainties being generated regarding track identity, position, course and speed due to the difficulties in classifying and localising targets at longer ranges.

4 Data from different sensors needs therefore to be combined in order to improve the confidence in track identity, position, course and speed. This requirement to use sensors co-operatively, dynamically reviewing the combination, create a significant challenge to aircrew.

5 Reduction in contact time for sensors results from continuing improvement in threat performance so that the window of opportunity for aircrew to detect, recognise and react to new contacts is diminishing.

The increasingly demanding role imposed on the aircrew and the associated time criticality associated with the necessary decision making based on assimilation, integration and interpretation of data from a multi-sensor mission system therefore lends itself to a knowledge based decision support solution.

For the Anti Submarine Role (ASW), the aircrew tasks (UK Observer, US Tacco) which could be addressed by applying KBS technology would be:

1 Deployment of Active Dipping Sonar and sonobuoy screening barriers

2 Active and passive location

3 Attack and re-attack

4 Lost contact procedures

5 Management of assets

Similarly in the Anti Surface Warfare role (ASuW) the aircrew tasks would be:

1 Classification of surface sensor data

2 Generation of surface picture based on classification

3 Path predicted for associated tracks

4 Plan area search routes

5 Assign contacts

6 Route production for confirmation of identity and hostility level of tracks

5.0 THE ASW/ASUW TECHNOLOGY DEMONSTRATOR PROGRAMME (TDP)

5.1 Background and Rationale

Other departments within DERA, particularly those associated with airborne and submarine surface sonars and anti air warfare in surface ships had also been examining and applying expert system technology. The Maritime Air Customer decided that a Technology Demonstrator Programme or US/ATD be mounted as a risk reduction exercise before considering the exploitation of Knowledge Based Decision Support technology in a mission system for the next generation of ASW/ASuW airborne platforms. To test the need for such decision support and to establish the breadth of functionality required structured interviews were conducted with authoritative sources of future operational requirements for maritime airborne platforms, DERA research sites and contractors engaged in developing workstation demonstrators. A functionality matrix was used incorporating a weighting schedule agreed by interview participants to establish need for and the functionality demanded of a decision support system to manage workload, maximise the use of mission systems and sensor resources to achieve consistency of mission performance.

5.2 Organisation and Components

Having agreed the need and functionality required for ASW/ASuW decision support the many workstation demonstrators developed under the sponsorship of DERA but targeted at specific areas of functionality were assessed for their relevance to the declared aim of satisfying the Maritime Air Customer requirement. A rainbow consortium of contractors had been formed in order to manage the intellectual property rights aspects and exploit the specialist experience of teams engaged in the wide range of small scale laboratory demonstrators. Representative threat scenarios were provided by the maritime Air Customer. The Rainbow Consortium included contractors with experience of building workstation demonstrators in the target domain, simulation environments to evaluate such demonstrators, data fusion systems and software to implement such schemes. These building blocks under consideration for the TDP included 7 separate workstation demonstrators, 3 simulation facilities, 2 computer simulation environments, 2 real time software toolkits and a data fusion system. Evaluation of building blocks was conducted using a selection strategy based on weighted requirements matrix including tasks defined in the approved scenarios together with functionality requirements. (Ref 11) This assessment having been achieved allowed the optimum architecture to be defined together with the associated components. Examination of maturity levels, flexibility and implementation considerations in association with a cost/benefit analysis led to the selection of the architecture and necessary components to achieve the core decision support system to implement the desired level of complexity to influence the Maritime Air Customer of the potential of the technology. This rigorous evaluation and trade off study resulted in the selection of an architecture which included:

1 A computer simulation environment, hosted on workstation which incorporated ASW and ASuW modelling capability.

2 A data fusion/association capability based on AAW Frigate TDP and ASW mission system.

3 The ASW and ASuW decision support laboratory workstation demonstrators.

4 The real time, multi agent software development toolkit D-MUSE due to its level of maturity; richer selection of programming strategies; represented the core of the key building blocks and cold be interfaced to the simulation environment.

5.3 Complexity

Further additional tasks are under discussion for inclusion by the contractors and the Maritime Air Customer in order to increase the capability of the ASW/ASuW Knowledge Based Decision Support Technical Demonstrator Programme. Man-in-the-loop evaluations are planned in order to demonstrate military worth of knowledge based DSS for the Maritime Air Customer. It can be seen that the nature and complexity of the tasks are very different from those usually encountered in the literature. A recent NATO KBS Working Group reported that KBS technology was sufficiently mature for applications in aeronautics and space due to their potential having been demonstrated in France, Germany and the USA in diagnostic and planning tasks in fielded systems. (Ref 12) More multi-function systems incorporating complex architectures were reported as being between the laboratory and fielded systems. The Maritime Air TDP described in this paper represents such an interim system.

6 0 THE THIRD MARITIME AIR APPLICATION - FUTURE ORGANIC AIRBORNE EARLY WARNING (FOAEW)

The MATDSS TDP represented a decade of research in software and tool development together with small scale ,piecemeal Maritime Air applications developed by different DERA departments and commercial organisations. The strategy for MATDSS was to use the building blocks systematically selected to create a TDP to demonstrate the potential military worth of decision support within a mission system for consideration for inclusion in a Naval Staff Requirement. A somewhat different strategy is being adopted for the Future Organic Airborne Early Warning platform[FOAEW] . The timescale available for developing the preliminary workstation demonstrator is 3 rather 8 years with the intention to move directly into a TDP. The AEW domain is significantly different from ASW/ASuW and possesses a far more demanding real time requirement. This was to be addressed by engaging the developers of the real time multi agent software and knowledge acquisition toolkit together with a Royal Navy AEW Instructor with direct access to AEW experts. The intention is therefor to build a comprehensive DSS from scratch rather than attempting to expand narrowly focused building blocks to attain a comprehensive decision support capability.

6.1 Overview of the AEW Task

A snapshot of a typical AEW scenario, in which AEW helicopters are responsible for protecting an advancing naval task force would include:

a) the task force being protected;

b) a frigate forward of the main force hosting the AEW aircraft;

c) a further frigate well advanced of the task force used as a weapons platform against incoming threats;

d) an AEW helicopter ahead of the task force searching for contacts and controlling the AEW operation;

e) fixed wing intercept aircraft well ahead of the force, in combat air patrols (CAPs) defined by the AEW helicopter, awaiting intercept request from the AEW helicopter;

f) further frigates on the task force flank used for ASW towed array operations (passive sonar); and

further helicopters ahead of the task force performing active sonar dipping in ASW operations. It is the task of the AEW helicopter to classify as early as possible all hostile airborne contacts in the vicinity of the task force, and to initiate the prosecution of those contacts it believes represents a threat to the task force, by efficient management of available assets.

6.2 Objectives of the Demonstrator

The purpose of this Decision Support System (DSS) is to develop a system which can be used to assess the expected increase in operational effectiveness of the Future Organic Airborne Early Warning (FOAEW) platform when operators are supported in their task by intelligent decision aids. From a successful demonstration of capability and improved effectiveness of operators, is the potential of moving the application toward a full mission system.

It is important to note that the proposed decision aid is not intended as an autonomous system with which the FOAEW operator has minimal interaction. Instead, it is required to be a co-operative system in which the system and operator are able to utilise the skills most appropriate to their capabilities. Even if it were technically feasible to provide an autonomous system, it is not obvious that this would be a desirable feature, since such a system would risk leaving AEW operators with no obvious control or responsibility over the progress of the sortie.

6.3 FOAEW DSS Scoping

The bases of the application scoping was the results obtained from many Knowledge Acquisition (KA) sessions. Each session focused on eliciting knowledge from an AEW expert with many years experience, and with knowledge of the likely evolution of present day AEW towards FOAEW. During this phase a semi-structured interview technique was employed in order to provide the necessary wide breadth of coverage of knowledge required for the scoping exercise. All sessions were recorded and transcribed (using the KA toolkit PCPACK, described later) providing a permanent record of the interviews, which will be available for reference throughout the development of the DSS.

The main roles for maritime forces in FOAEW were identified as Surveillance, Attack Co-ordination, Airmanship (including self-defence and safety) and Communications. Tasks in support of these capabilities include radar and sensor handling, detection, situation assessment, tactical decision making, fighter control, reporting and airmanship. For each of these tasks different levels of support were identified, ranging from routine activities to those requiring active intelligent processing. Eleven possible areas of support for the FOAEW tasks were identified:

Barrier Positioning

CAP Control

Radar Sensor handling

Tactical Decision Making

Prosecution and Attack Co-ordination

Reporting and Data Link Management

Database management

Airmanship

Self defence and Safety

Secondary Roles

These provide the sub-division of the major functions likely to be involved in performing FOAEW. For each of these areas, specific tasks were identified in which a DSS could provide operator support.

7.0 DESIGN METHODOLOGY AND KA TOOLS

Although the initial scoping phase utilised semi-formal interview techniques, the current design phase is employing a more formal approach using the KA toolkit (PCPACK).

7.1 KA Toolkit: PCPACK

Essential to the design process, significant benefits accrue from using an appropriate toolkit in support of KA. For the FOAEW DSS knowledge acquisition is being supported by the KA toolkit PCPACK [Ref 3]. PCPACK consists of an extensive collection of KA tools, KADS style directive models, and facilities for supporting project management and documentation. These include:

- GDM Workbench;
- Protocol Editor (or Transcript Analysis tool);
- Hyperpic tool;
- Laddering tool;
- Matrix tool;
- Card Sort tool;
- Repertory Grid tool;
- Case Editor and Rule Induction tool;
- Rule Editor;
- Project Management; and
- Project Documentation.

The KA toolkit binds the various tools together using an object database, within which the instances of concepts derived by each tool, and their inter-relationships, are stored. The integrated database allows the different facets of the stored knowledge to be viewed and manipulated by other tools in the toolkit.

7.2 The Generalised Directive Model (GDM)

An important part of the design process using PCPACK is the identification of the GDMs (Generalised Directive Models), for the tasks of the application.

The GDM comprises a statement of the inference steps performed during problem solving (for example, abstract, match and refine), the classes of domain descriptions serving different problem solving roles (PSR) within the model (for example observables, variables, abstract solutions and specific solutions) together with the dependencies between the two.

These inference steps and roles correspond to partitions in the knowledge of the system. Aiding acquisition and the organisation or indexing of the knowledge required for the system to function.

For example, a directive model for performing situation assessment could be to match the known features of a particular contact (track) with typical descriptions of certain types of objects (schemata).

7.3 Detailed KA for FOAEW

All of the KA carried out during the detailed design phase for the FOAEW DSS is being captured using the PCPACK toolkit, which will be used to provide a knowledge model for each of the tasks identified during the scoping phase.

The design process is often found to be iterative, since the GDM may be unknown until an initial examination of the task has been carried out. Subsequently, an appropriate GDM may be identified but it is found that it does not completely match the task decomposition. In this case it may be desirable to reassess the task to see if it can be recast to match the GDM more closely.

8.0 IMPLEMENTATION TOOLKIT: D-MUSE

The proposed implementation framework for the FOAEW DSS is D-Muse [Ref 19], a real-time knowledge based toolkit for the development of distributed applications.

D-Muse provides facilities for running a collection of named processes, that are inter-connected by a network of deadlock free communication paths.

Each Muse process provides the following functionality:

- *flexible knowledge representation*:
- *support for modular code development*:
- *support for real-time operation*:
- *extensible*:
- *distributed object management*:
- *session management*

D-Muse is not restricted to inter-connecting Muse processes, but can also be used to communicate with other processes, such as graphical user interfaces and simulation facilities.

9.0 ARCHITECTURAL DESIGN

Another important goal for the FOAEW DSS is that it should provide an extensible implementation that supports, where possible, the reuse of components, and the ability to grow the system without having to re-engineer components. To achieved this, an agent based architecture has been chosen, which is being implemented within D-Muse.

The internal architecture for the agent is being designed so that they can be populated directly from the knowledge models created during the design phase.

Analysis of the task decomposition and its GDM will identify the requirements and capabilities of the agent. In particular, elements of the world model will be defined by the input and output parameters to the GDM. Requirements and capabilities will be broadcast (handled by a *facilitator* agent) to the other agents of the system. The other agents are able to respond with offers of appropriate information, or with requests to utilise the capabilities offered by the agent.

For example, the identification of Missile Engagement Zones (MEZs), important for safe route planning, can come from many different sources, some of which may not be implemented in the initial DSS. The addition of further agents that expand the scope of MEZ identification is facilitated by the autonomy offered in an agent architecture.

The following diagram, figure 1, shows our initial application architecture.

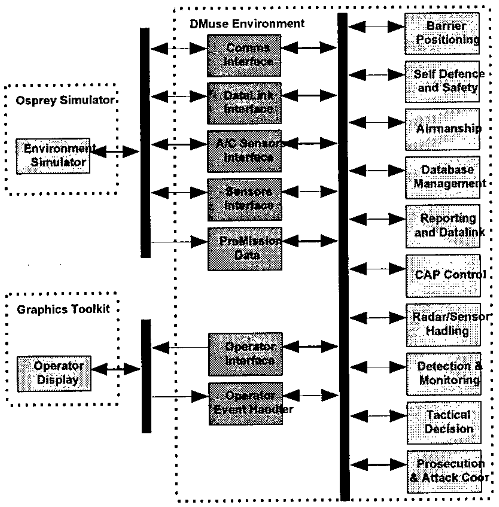


Figure 1: Key Software Components

The agents on the right represent the kernel functionality of the system, the central set of agents define the support agents that interface with, in our DSS, a simulator and a workstation based operator interface. This decomposition is not definitive. Changes are already being considered to partition a number of the agents into smaller agents, with more specific responsibilities, which then report to a controlling managing agents.

10.0 AGENT COMMUNICATION LANGUAGE

The agent communication language chosen for the FOAEW DSS is the Knowledge Query and Manipulation Language (KQML). KQML is a language independent message format and message-handling protocol designed to support run-time knowledge sharing among agents.

For Muse implemented agents (this will apply to most of the agents within the DSS), only the semantics of KQML will be implemented.

However, where there is a need to communicate with agents that have not implemented in the Muse language, or do not support D-Muse streamed object communication, the full syntactic form of KQML messages will be employed.

This approach will aid future desires to integrate the DSS with legacy systems, such as intelligence and emitter databases.

10.1 Optimisation of Inter-agent Communication

An agent defines a logical partition of functionality within the application, but does not require that it be localised to a single processor. A proposal is currently being investigated of ways of distributing agent functionality across processor boundaries to find ways of improving efficiency of inter-agent communication in a distributed environment.

In terms of its implementation it is envisaged that this will involve sharing elements of the world and acquaintance models across processor boundaries (implemented using D-Muse mirrored objects).

10.2 Fault Tolerance and Agent Mobility

Since the proposed D-Muse agents can be created dynamically across any of the available CPUs, the run-time system provides capability for implementing a limited degree of fault tolerance within the DSS. For example, if a processor dies it is possible to terminate the function of low priority tasks, and use the freed resources to host new instances of agents from the lost processor.

11.0 CURRENT STATUS OF THE FOAEW DSS

The methodology, design tools and the proposed agent architecture for the FOAEW DSS provides the developers with confidence in the development of the application.

In particular, the close relationship between the methodology, it realisation in the PCPACK toolkit and the ability to map the knowledge concepts directly onto features of the implementation language, provide a means of estimating the size of the application (including the size and quantity of data structures, and the complexity of the various functional areas and tasks of the application).

The utilisation of an agent architecture, although grounded more in practical implementation rather than a theoretical implementation, provides the developers with means of reusing software components and supporting future modular expansion.

The use of KQML style communication between agents of the application, allowing easily transition into the KQML syntax for communication with external data sources that will grow in importance as the system expands.

12.0 CONCLUSIONS

In a recent journal article, Hayes-Roth, drawing on two decades of research experience for DARPA, contended that much has been achieved using AI techniques in an incremental fashion in relatively small scale exercises. (Ref 13) However effort needed to be concentrated over a period of time in specific domains to demonstrate potential before adapting and transferring solutions to new situations. Context adaptable building blocks, re-useable knowledge and composite architectures for multi task systems were recommended as necessary constituents of a future strategy for AI. The authors' of this paper contention would be that in concentrating on Maritime Air ASW (Ref 14), ASuW, (Ref 15), AEW (Ref 16) for almost a decade, and in developing the necessary support software tools and methodologies, together with the re-use aspects, the KBS Group within the DERA organisation in the UK already conform to most aspects of the Hayes-Roth paradigm.

The paper describes two contrasting approaches for the building of TDP'S. Each possesses its own merits and as both have yet to be completed, the relative merits of different approaches have yet to be demonstrated and assessed. Both represent a complexity far beyond that referred to in the recent US survey reported in AI Magazine (Ref 13) and those cited in the NATO/AGARD report (Ref 12). MATDSS encapsulates a slowly developing scenario whilst FOAEW demands a much faster tempo of response. The real time software and knowledge acquisition toolkits used are commercially available products. Both are attempts to demonstrate the military benefit of incorporating decision support in airborne mission systems.

A recent US survey indicated that the KBS tool and consultancy market within North America was $258 Million but reported that the activity in the UK remains conservative in development and deployment of expert systems. Maritime Air is certainly an exception to this generalisation but awareness is no doubt limited due to exposure being confined to forums such as this conference.

The expertise accumulated in over a decade of research by DERA is now being expanded into the exploration of the potential of utilising non symbolic AI, rather than symbolic AI, in mission systems associated with unmanned vehicles.

References

1 HOWELLS, H. BICKERTON, R. Lt (RN). *Decision making in ASW helicopters using KBS advisors.* RAeS Conference 'Role of Intelligent Systems in Defence' at St Hugh's, Oxford Mar 1995.

2 MERRIMAN,S. edit. *"Applications of System Ergonomics to Weapon System Development"*, NATO-DRG Panel 8 Workshop (Unpublished) RMCS Shrivenham 1984

3 PROF SHADBOLT, N. et al. PC PACK Release Notes (Version R 1.0) October 1995

4 MARTIN, S., HOWELLS, H. *"Real Time Software for Knowledge Based Systems"*, IE^3 Colloquium London 1995

5 GRISONI, M. HOWELLS, H. *European Workshop on Verification and Validation of KBS.* Proceedings of EUROVAV 91, Jesus College, Cambridge, July 1991.

6 GRISONI, M. VORTEX Final Report Logica CAM.706.70015-FRC Sept 1986

7 HOWELLS, H., PEDEN, C G. Lt (RN) *Tactical Decision Aid for ASW Aircraft* DRA Report Jan 1992 (Unpublished)

8 SMALL, Major R. USAF *"The Pilot's Associate - Today and Tomorrow"* Proceedings of the 1st USAF EAORD International Workshop on Human - Electronic Crew, Ingolstadt, Sept 1988.

9 DOE, E. et al. *Applied Study into KBS for Fixed Wing Aircraft* FARL Report 262/224 March 1987

10 ZANCANATO, R. HOWELLS,H. *"An Agent Based Helicopter Decision Support System"*, British Computer Society Expert Systems Conference Proceedings, Cambridge December 1995

11 BENTLEY, P. et al *Need and Functionality of a Maritime Aircraft TDSS* Logica Report February 1996

12 PROF WINTER, H. edit *"Knowledge Based Guidance and Control Functions"* NATO AGARD - AR-325 1995

13 HAYES-ROTH, F. *AI What works and what doesn't.* AI Magazine V18 N2 Summer 1997

14 ELLIS, R. *A report on the design of Tactical Decision Aids for Maritime Aircraft operating in a multi-platform co-operative environment.* DRA Report Nov 1996 (Unpublished)

15 DAVIES, A J Lt (RN)., HOWELLS, H. *KBS Mission Manager Concept Report* DERA Report May 1997 (Unpublished)

16 HOWELLS, H., DAVIES, Lt A.J. RN. *Recommendations for the application of KBS Decision Support technology for the FOAEW platform* DERA Report Nov 1996 (Unpublished)

APPLICATIONS OF ARTIFICIAL NEURAL NETWORKS AND GENETIC ALGORITHMS TO ELECTROMAGNETIC TARGET CLASSIFICATION

G. Turhan-Sayan
S. İnan
T. İnce
K. Leblebicioğlu

Electrical and Electronics Engineering Department
Middle East Technical University
06531 Ankara, Turkey

SUMMARY

This paper presents two approaches for electromagnetic target classification which utilize learning, self-organizing and evolutionary algorithms for data processing.

The first approach to be discussed here is based on artificial neural networks where either a feed-forward network (a multi-layer perceptron) or a self-organizing map can be used as the main building block of the target classifier that must also contain a special signal processing unit for feature selection and/or feature enhancement. Based on the simulation results to be summarized, a modified self-organizing map supported by a Wigner distribution type two-dimensional signal processing unit has been found to exhibit an excellent classification performance.

The second target classification approach outlined in this paper describes an ultra-wide band classifier based on the annihiliation of natural resonances of radar targets. Use of genetic algorithms are found to be invaluable in the design of the target-specific filters characterized by special time-limited signals.

1. INTRODUCTION

Accurate and fast recognition of various targets such as aircraft, missiles, ship and land vehicles using their scattered field data is an important problem especially for military applications of radar target classification. As known, all the characteristic information about size, shape and electrical properties of a target is implicitely contained in the scattered signals received from the object. However, such data are strongly dependent on polarization and aspect angle of transmitted and received radar signals as well as the measurement band of frequencies. For that reason, the database needed by a target classifier must contain scattered signatures for each candidate target within a specific class over a proper bandwidth at many combinations of aspect angle and/or polarization to provide sufficient characterization. A feasible target classifier must be based on a proper data processing technique which does not require too much time for real-time processing to deal with such a huge volume of data. This constraint calls for extensive passive time data processing prior to the active target recognition phase. The results of our recent research studies in this direction are presented here while focusing on two different target classification approaches.

The first approach makes use of *artificial neural networks* (ANN) together with *time-frequency signal representation* (TFR) techniques for target classification. Lately, ANNs have been utilized in various engineering applications including the electromagnetic target classification problem due to their ability to learn and generalize as well as their capacity for massive parallel processing. As the *multi-layer perceptron* (MLP) type ANN applications in radar target classification have already been discussed thoroughly in literature [1-3] , our MLP classifier results will not be repeated here. Instead, the original application of *self-organizing map* (SOM) type ANNs to target classification will be described in section 2.1 .

It is well known that the performance of a target classifier strongly depends on the properties of the database used in the process. A basic database in our present problem is composed of radar returns from the candidate targets at various aspect angles and/or polarizations either in time domain or in frequency domain. In most cases, however, this essentially unprocessed database is not useful enough to achieve acceptable classifier performance. Some sort of feature extraction from this basic database is needed to eliminate similarities and to emphasize fundamental differences between the targets. That helps the ANN classifier to yield higher correct decision rates in the expense of some increase in computational work. It should be indicated in this context that the *Wigner distribution* (WD), a quadratic time-frequency representation, is utilized for feature extraction in our simulations as explained in section 2.2. Results of a computer simulation for a classification group of five conducting model aircraft using a SOM type ANN classifier are given in section 2.3.

In section 3 of the paper, theoretical aspects and simulation results for a wide-band resonance-based target recognition approach are presented. In section 3.1, the theoretical background of the technique will be summarized based on the natural resonance annihilation principle. Next, in section 3.2, the use of *Genetic Algorithms* (GA) to synthesize target-specific input pulses will be explained. Results of a related simulation problem for a classification group of three linear thin conducting wires will be presented in section 3.3.

Paper presented at the RTO SCI Symposium on "The Application of Information Technologies (Computer Science) to Mission Systems", held in Monterey, California, USA, 20-22 April 1998, and published in RTO MP-3.

2. TARGET CLASSIFICATION WITH NEURAL NETWORKS

As briefly discussed in the introduction, a typical radar target classification problem involves a large database of scattered signals most of that should be processed prior to the *real-time* or *active* classification phase to ensure an acceptable classifier speed. In other words, whenever possible, all useful information such as the individual target features or certain similarity measures between the targets within a given class must be extracted from the available database before the test signal from an *unknown target* is received. Hence, the active classification phase can be reduced to a relatively simple comparison stage between the extracted features of the test target and the already extracted features of the whole class. In an ANN based target classifier, this comparison stage as well as part of the information filtering stage can be successfully implemented by certain learning, self organizing and testing algorithms.

2.1 Theory

An artificial neural network is a signal processing device which is composed of a large number of interconnected parallel processing elements called *neurons* [4]. Learning in an ANN is accomplished by exposing the network to training inputs which are randomly selected from an available *training database*. During this process, the ANN adapts itself to the training information environment by updating the weight vectors associated with neurons. The network topology, neuron characteristics and learning rules determine the type of an ANN. According to their learning strategies, ANNs can be classified as *supervised* and *unsupervised* networks. In supervised learning, each time the ANN is exposed to a training input, the related class information (i.e., the expected ANN output) is provided as well. The multi-layer perceptron (MLP), the feedforward ANN as also called, has been the most popular example to the supervised neural networks. The well-known back-propagation algorithm used for training MLPs is an iterative gradient descent technique based on the minimization of error between the current and the expected network outputs. The self organizing map (SOM) type ANN, on the other hand, is an example to unsupervised neural networks which do not need any class information for learning but acquire that knowledge by itself during the training phase through cluster formation.

Introduced for the first time by Kohonen [5], the SOM algorithm creates a mapping from a high dimensional input vector space onto a two dimensional output lattice and is basically composed of a single, two-dimensional layer of neurons. Associated with a neuron $n_{i,j}$ of this layer, there exists a weight vector $\vec{w}_{i,j}$ which is of the same dimension as the input feature vector $\vec{x}$. The SOM algorithm is initialized by assigning random and usually small values to all these weight vectors. At each iteration of the SOM training phase, a randomly selected input feature vector is applied to each and every neuron of the output lattice. Then, the neuron at the lattice location (i^*, j^*), whose weight vector $\vec{w}_{i^*,j^*}$ best approximates the input feature vector is chosen as the *winning neuron* according to the following rule:

$$\underset{\mathbf{i,j}}{\textbf{Minimize}} \quad \left\| \vec{x}(t) - \vec{w}_{i,j}(t) \right\| \tag{1}$$

where the norm is computed as

$$\left\| \vec{x}(t) - \vec{w}_{i,j}(t) \right\| = \sqrt{\sum_k [\, x_k(t) - w_{i,j,k}(t)\,]^2} \tag{2}$$

with k being the dimension index of the input and the weight vectors, and t being the iteration index. Then, the weight vectors of the neurons are updated to be used in the next iteration as

$$\vec{w}_{i,j}(t+1) = \vec{w}_{i,j}(t) + \mathrm{N}_{i,j}(t)\,\eta(t)\,[\,\vec{x}(t) - \vec{w}_{i,j}(t)] \tag{3}$$

where $N_{i,j}$ (t) is a neighborhood function centered around the winning neuron n_{i^*,j^*} and η(t) is the learning rate. As the training phase progresses, the learning rate is gradually decreased and also the neighborhood function is made better localized (narrower) about the winning neuron to provide finer tuning. The neighborhood function can be chosen as a two-dimensional pulse function (step pulse, gaussian pulse, mexican-hat shape pulse, etc.) to affect the neuron weights only or mostly in a subregion of the SOM lattice centered at the winning neuron. Iterations are continued until all the weight vectors are stabilized and hence cluster regions for each candidate target are properly established on the SOM output lattice. The resulting output lattice is called the SOM *output map*.

2.2 Feature Extraction Using the Wigner Distribution

A suitable feature extraction/enhancement procedure applied to a given input database certainly helps to improve the correct classification rate in target recognition applications. In the present problem, the input vectors, $\vec{x}$, fed to the SOM algorithm, are obtained after performing feature extraction by means of a time-frequency signal representation technique. The *Wigner distribution* (WD), which essentially produces an energy distribution map with respect to time and frequency (simultaneously) for a given target signature, is used for that purpose in our work [3,6]. The auto Wigner distribution of a continuous time signal x(t) is defined as

$$W_x(t,f) = \int_{-\infty}^{+\infty} e^{-j2\pi f t} x(t+\tau/2)\, x^*(t-\tau/2)\, dt \tag{4}$$

where the symbol "*" represents the complex conjugate operator.

In particular, the WD satisfies two important expressions called *time-frequency marginals* stated as

$$\int W_x(t,f)\,df = |x(t)|^2 \tag{5.a}$$

and

$$\int W_x(t,f)\,dt = |X(f)|^2 \tag{5.b}$$

where $x\ (t)$ is the time signal with $X(f)$ being its spectrum, $|x(t)|^2$ is the instantaneous power of the signal at a given time t and $|X(f)|^2$ is the spectral energy density of the signal at a given frequency f. The total energy E_x of the signal can then be computed from

$$E_x = \iint W_x(t,f)\ df\ dt \tag{6}$$

Similarly, the energy contribution of the spectral terms in the frequency range $[\,f_1\,, f_2\,]$ to the time interval $[\,t_1\,, t_2\,]$ can be estimated from

$$E_{partial} = \int_{t_1}^{t_2}\int_{f_1}^{f_2} W_x(t,f)\,df\ dt \tag{7}$$

Based upon these properties, it can be deduced that for given pairs of time and frequency (t,w), an energy distribution type feature matrix can be constructed over the two-dimensional time-frequency domain using the WD for each scattered signal in the classification database.

2.3 Application

The aim of this simulation example is to classify targets within a group of five model aircraft using a SOM classifier together with WD type feature extraction. The targets are metal electroplated small-scale versions of the planes Boing-707 (target A), Boing-727 (target B), Boing-747 (target C), Concorde (target D) and DC-10 (target E). The vertically polarized frequency domain backscattered data for all five targets (at various aspect angles) were measured at the Ohio State University compact RCS measurement range over the frequency band [1GHz-8GHz] with 0.05 GHz steps. The corresponding impulse response waveforms for the targets were obtained by taking the IFFT of the windowed frequency domain data. The available unprocessed time-domain database (i.e. the database before feature extraction) is composed of these impulse response waveforms. Part of this database is used to train the SOM algorithm and the remaining part is reserved for testing purposes. The simplified block diagram of the target classifier used in this example is shown in Figure 1.

As discussed earlier, most of data processing required for target classification must be completed in the *passive* classification phase to minimize the amount of computational tasks to be performed in real-time. The first step of this phase in our work is feature extraction by the WD approach to obtain energy distribution feature matrices for each time-domain backscattered signal in the database. Next, sufficiently late-time portions of the Wigner distribution output matrices are selected to further emphasize the natural resonance behaviour of the targets. After computing partial signal energies in certain late-time slots at each sample frequency (used for WD matrix computations) , energy feature vectors are obtained to be fed to the SOM classifier. To be specific, for each signal of the database, a WD output matrix is computed first of all for 1500 time sampling points in the time range [0,7.32 nsec] and for 50 frequency sampling point in the frequency range [-4 GHz, 4 Ghz]. Then, the time range is divided into ten equally wide time slots of 0.732 nsec each. The WD outputs are integrated over time within each time slot at each computation frequency to obtain energy distribution patterns of length 50. The patterns corresponding to sixth, seventh and eighth time slots are combined to form an energy distribution feature vector of length 150 representing the late-time behaviour of the original signal. The impulse response waveforms and the related energy feature vectors for the targets B-707 (target A) and DC-10 (target E) at 45 degree monostatic aspect angle (measured from the nose of the planes) are shown in Figure 2.

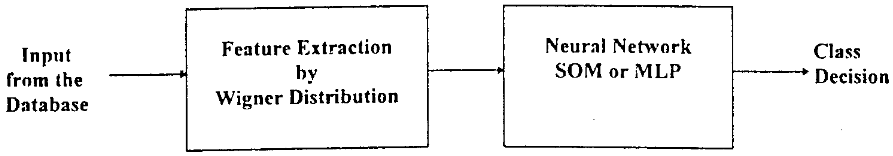


Figure 1. Block diagram of the ANN based target classifier

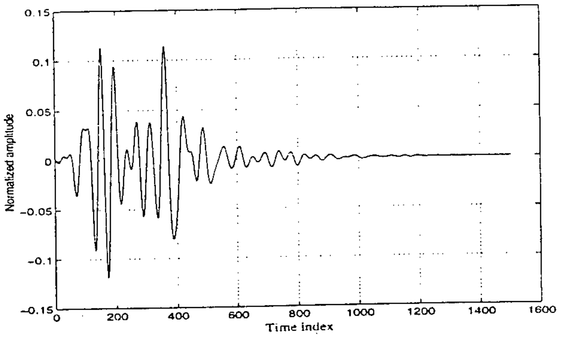


Figure 2.a Normalized impulse response for target A at 45 degree aspect angle, vertical polarization.

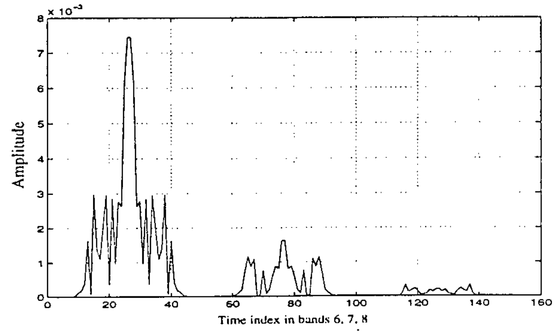


Figure 2.b Energy feature vector for the database signal shown in part (a).

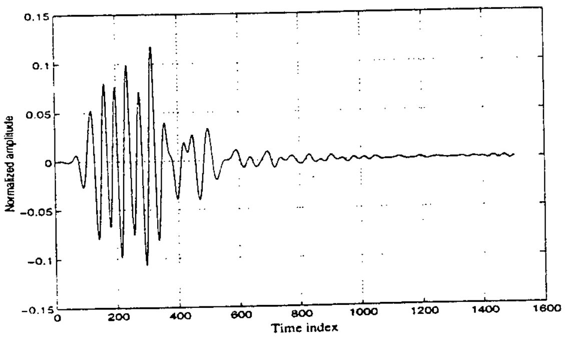


Figure 2.c Normalized impulse response for target E at 45 degree aspect angle, vertical polarization.

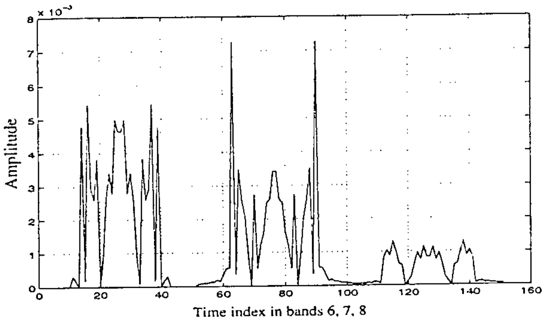
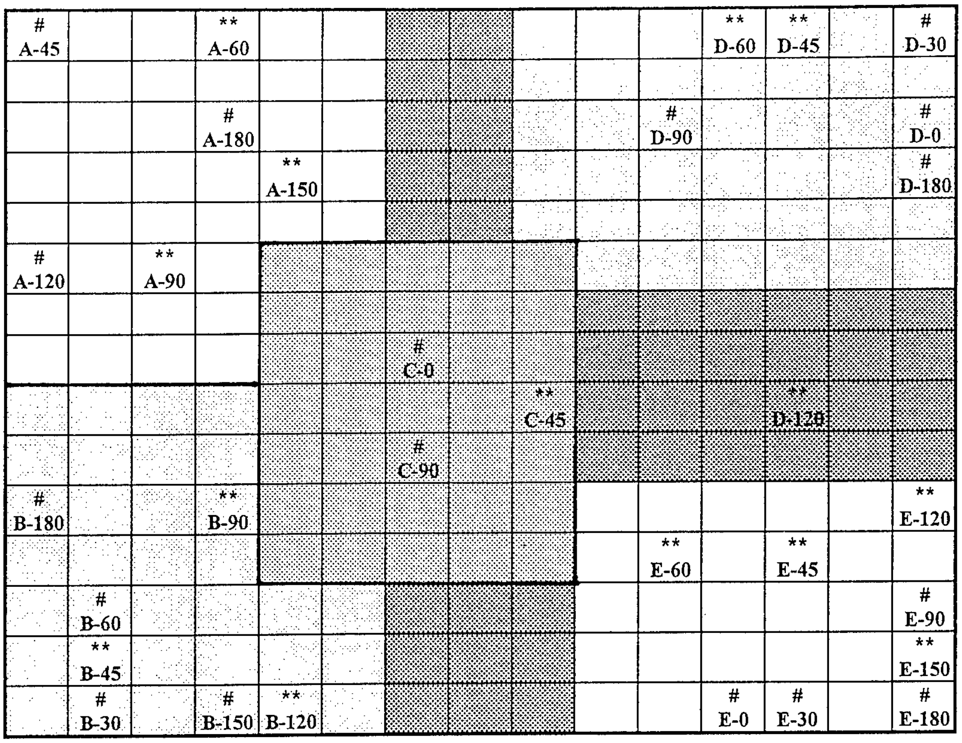


Figure 2.d Energy feature vector for the database signal shown in part (c).

The second step of the passive classification phase is the training of the SOM type ANN using a set of energy feature vectors characterizing the targets of the classification group. As discussed earlier, training of the SOM is equivalent to establishing a weight vector for each SOM neuron such that non-overlapping cluster regions (one for each airplane) on the SOM output map are formed. A proper cluster region must contain all the winning neurons (obtained at various aspect angles) for only one of the targets in the classification group. Then, in the active recognition phase, the input feature vector of an unknown test target can be simply compared with the SOM weight vectors to determine a test winning neuron . The label of the cluster containing this test winning neuron identifies the target. When the standard SOM algorithm was first applied to our electromagnetic target classification problem, a problem was encountered in forming proper non-overlapping cluster regions on the output map. The problem was mostly due to the high degree of similarity between the geometrical features of the perfectly conducting model airplanes that at some aspect angles used in training, two or more airplanes might have quite similar backscattered signatures. As a solution, we added some supervision to the SOM algorithm to guide and accelerate the cluster formation by declaring winning neuron locations for such difficult cases[3,7]. In other words, the original unsupervised SOM algorithm has been modified in the training phase while the test phase of the classification was still allowed to follow the standard SOM rules.

Numerically, a SOM output lattice of size 15X15 is utilized with a total of 225 neurons. Each neuron is associated with a weight vector of length 150 that is the same length for the input feature vectors. The learning rate of the training algorithm was set to 0.2 initially and decreased gradually as training proceeded. A two-dimensional unit pulse was chosen as the neighborhood function whose spot size was large enough to span the whole lattice initially but was reduced gradually during the training process. Maximum



# A-45			** A-60								** D-60	** D-45		# D-30
			# A-180							# D-90				# D-0
				** A-150										# D-180
# A-120		** A-90												
						# C-0								
								** C-45				** D-120		
						# C-90								
# B-180			** B-90											** E-120
										** E-60		** E-45		
	# B-60													# E-90
	** B-45													** E-150
	# B-30		# B-150	** B-120							# E-0	# E-30		# E-180

Figure 3. SOM output map of 225 neurons showing cluster regions for targets A, B, C, D and E where darkly shaded regions do not belong to any cluster region (# shows training winning neurons, ** shows test winning neurons)

number of training iterations were set to be 30 000 and 17 of the 31 available database signals were used for training the SOM. The remaining 14 signals were utilized for testing. Results of SOM training and testing are presented on the SOM output map shown in Figure 3. The winning neurons established during the training phase are designated by the symbol '# ' and the winning neurons of the testing procedure are indicated by the symbol '**' on this map. The cluster regions belonging to targets A, B,C,D and E are shown in Figure 3 as well. Based upon the information obtained from this figure, the correct classification rate is 93 % with 13 correct classifications out of 14 test cases. It is observed that the testing winning neuron representing the data of Concorde (target D) at 120 degrees aspect angle is the only one falling in a neutral region (shaded in the darkest color in Figure 3) leading to not an incorrect but an ambiguous classification result.The cpu processing time for this classifier simulation was measured approximately as 250 msec (including about 190 msec spent for the feature extraction stage) on a Sun Sparc 4 machine.

3. THE USE OF GENETIC ALGORITHMS IN RESONANCE-BASED TARGET CLASSIFICATION

The electromagnetic target classification technique to be discussed in this section is basically a signal shaping approach based on the annihilation of a target's natural resonances. Use of optimization techniques are required in the passive recognition phase (prior to receiving the test signal from an unknown target) for the design of characteristic pulse signals one for each candidate target in a classification group. The genetic algorithm optimization is the most useful tool for this purpose especially when there is no information available for the pulse duration.

3.1 Theory and Formulations

Use of input signal shaping in electromagnetic target identification problems was suggested by E. M. Kennaugh, in his 1981 paper [8] where he introduced the K-pulse concept for the first time as well. Kennaugh defined a special input signal, called K-pulse, as a target-specific and time-limited excitation signal to produce time-limited scattered responses at *all* possible combinations of aspect angle and polarization. Due to its compact support, the Laplace transform of a K-pulse is an entire function of complex frequency having no poles. The set of zeros of this entire function is required to be the same as the set of complex natural resonance (CNR) frequencies (i.e., system poles) of the related target to have time-limited target responses at all possible aspects and polarizations. As the accuracy of extracting CNR frequency information from noisy experimental data is very low, K-pulse shaping can not be realized as a simple zero-insertion process in the Laplace domain. Instead, the K-pulse synthesis procedure must be based on an optimization problem where the cost function to be minimized is the natural resonance-related late-time energy content of the scatterer response [9,10]. For simple canonical target geometries, the pulse duration can be estimated [11]. Then, the K-pulse signal can be expanded into a proper basis over a specific time interval and the expansion coefficients optimized for minimum cost define the shape of the pulse. The classical gradient-search type optimization algorithms are found to be adequate in solving this problem. For complicated target geometries, however, there is no well-established rule for guessing the K-pulse duration that is the most critical parameter of the signal shaping procedure. In that case, not only the shape but also the duration of the K-pulse signal must be optimized and it is an extremely difficult optimization problem. As discussed in [12], classical optimization algorithms are found useless in solving this challenging problem as they easily get trapped at local minima of the cost function unless very good initial guesses, especially for the signal duration, are provided. The genetic algorithms, on the other hand, are known to be quite successful in finding the global solution to complicated optimization problems without any need for good initial guesses.

The formulation of the K-pulse synthesis problem can be summarized as follows: As discussed in references [13] and [14], a finite-size perfectly conducting scatterer can be approximately represented by a linear, time-invariant system model. The scattered response, $r^i(t)$, of the target to an arbitrary input signal x(t) applied at the i^{th} combination of the aspect angle and/or polarization is expressed as

$$r^i(t) = x(t) * h^i(t) \tag{8}$$

where $h^i(t)$ is the scatterer's impulse response and the asterisk " * " denotes the convolution operator. As discussed in [15], the K-pulse signal, k(t), can be expanded into a modified Legendre polynomial basis as

$$k(t) = \sum_{n=0}^{\infty} D_n \overline{P}_n(t) \tag{9}$$

where

$$\overline{P}_n(t) = \sum_{m=0}^{n} \alpha_{nm} \left(\frac{2}{T_k} t - 1\right)^m \tag{10}$$

is the n^{th} order modified Legendre polynomial defined over the interval $[0, T_k]$ with T_k being the K-pulse duration and α_{nm} ' s are the tabulated construction constants for the original Legendre polynomials which are defined over the interval [-1,1]. D_n ' s are the expansion coefficients to be optimized. If a target is excited by its own K-pulse signal, the resulting K-pulse response , $r_k^i(t)$, at the i^{th} combination of aspect and polarization, is given as

$$r_k^i(t) = k(t) * h^i(t) \tag{11}$$

and as implied by the definition of K-pulse,

$$r_k^i(t) = 0 \qquad \text{for} \qquad t \geq T_{L_i} = \frac{L_i}{c} + T_k \tag{12}$$

where L_i is the target dimension along the line of sight at an aspect angle θ_i and c is the speed of light in air. In other words, the span of the target's K-pulse response, T_{L_i} , must be equal to the span of its forced component as the natural response component is expected to be annihilated under the K-pulse excitation. Then, the K-pulse synthesis problem can be posed as follows :

$$\textit{Minimize } J(T_k, D_0, D_1, \cdots) = \sum_{i=1}^{I} W_i J_i \tag{13}$$

where

$$J_i = \int_{\frac{L_i}{c}+T_k}^{\infty} [r^i(t, T_k, D_0, \cdots)]^2 \, dt \tag{14}$$

with respect to the K-pulse duration, T_k , and the expansion coefficients, D_n ' s . The cost function J represents the weighted sum of the natural response-related late-time energy contents, J_i , of the target response at I different combinations of aspect and/or polarization. The W_i terms represent the properly chosen weight factors if a multi-combinational K-pulse synthesis is needed in the case of complicated target geometries. For simple target geometries, even I=1 would be sufficient. In general, this target classification technique needs wide-band data measured at only a few different aspects and/or polarizations.

After the K-pulse signals for every target in a classification group are synhesized properly, the active (real-time) target classification phase can be accomplished by convolving the impulse response of the *unknown target* with each and every K-pulse signal already stored in the computer. In this process, each K-pulse signal represents a digital filter, in a sense, as shown in Figure 4. The K-pulse filter having a time-limited output with the shortest time span determines the class of the test target. As the cpu time spent for a discrete convolution is extremely short, the target classification technique based on K-pulse concept turns out to be very fast.

3.2 Implementation of Genetic Algorithms

Genetic algorithms (GAs) are adaptive parallel search techniques of probabilistic nature and have been inspired by the genetic evolution processes [16,17,18]. The most important advantage of the GAs is that they are designed to search for the global optimum without the need for good initial guesses. Furthermore, as GAs do not require gradient information, a lot of flexibility is provided in cost function selection. Due to these facts, complicated optimization problems which can not be solved by

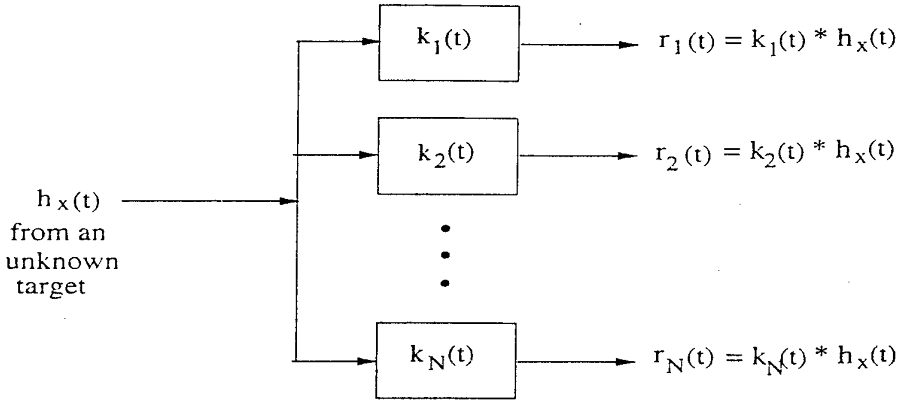

$k_1(t)$
$r_1(t) = k_1(t) * h_x(t)$
$k_2(t)$
$r_2(t) = k_2(t) * h_x(t)$
$h_x(t)$ from an unknown target
$k_N(t)$
$r_N(t) = k_N(t) * h_x(t)$

Figure 4. Block diagram for the K-pulse based target classification scheme

traditional optimization techniques can be successfully handled by GAs.

A GA basically manipulates a fixed-size population of encoded structures, called *chromosomes,* through the application of genetic operators which are *selection, crossover* and *mutation.* A chromosome can be encoded as a string of binary digits and represents a vector of optimization variables in the feasible solution space. Each chromosome is associated with a measure of performance called *fitness*. Chromosomes with higher fitness values have greater chances of reproduction and survival in the population. As the operation of a GA is based on fitness maximization, a proper mapping from the cost (J) to be minimized to a fitness (f) is needed. The first step of a GA is the random generation of the initial population of chromosomes. Then, the crossover, mutation and the selection steps are performed in succession to obtain the population for the next generation.

The *crossover* operator works on randomly selected pairs of chromosomes from the current population (with a given probability called *crossover rate,* p_c) to generate offsprings by interchanging segments of parent chromosomes at randomly chosen crossover points along the binary strings. The idea behind the crossover operation is to make the cross-breeding of good solutions possible so that their best components may be combined to form even better solutions. The mutation operator, on the other hand, acts on the offspring chromosomes to toggle the bit values with a very small probability called *mutation rate,* p_m. This operator randomly adds new genetic material to the population to enhance the possibility of finding a better solution. The final step of each generation cycle is the selection of the chromosomes for the next generation from the collection of the parent and offspring chromosomes. The basic idea behind the selection operation is to keep the chromosomes with high fitnesses and eliminate those with low fitnesses to converge to the optimum solution while maintaining a reasonable level of randomness in selection to avoid the local optima.

3.3 Application

The use of GAs in K-pulse shaping is demonstrated in this section for a perfectly conducting cylindrical thin wire of 12 meters long with the length-to-diameter ratio, (L/d), of 2000. The ϕ-polarized synthetic backscattered data for the wire at the aspect angle of 30 degrees (measured from the wire axis) is computed by a moment method algorithm over the frequency range 2 to 256 MHz. The impulse response required at this single synthesis aspect is obtained by computing the inverse FFT of the available frequency domain data. The K-pulse for the wire is modeled as shown in Equation 9 and the cost function given by Equation 13 is minimized to obtain the K-pulse duration, T_k, and the expansion coefficients, D_n's. The modified Legendre polynomial basis is truncated to include only the first six lower order terms as the expansion coefficients for the higher order terms are observed to have negligibly small values. Each optimization variable is coded by 10 bits in the intervals indicated in Table 1 leading to a chromosome length of 70 bits. The population size is chosen to be 150 chromosomes. The crossover rate and the mutation rate are taken as 0.95 and 0.05, respectively. Also, a fitness mapping to the interval [100, 101] is carried on for improved GA performance. The Roulette Wheel Parent Selection technique, one crossover point per variable, Steady-State Without Duplicates Reproduction technique and Elitism are employed in the GA algorithm. The resulting K-pulse of the test target is obtained in less than 20 iterations and plotted in Figure 5. The optimized variable values are presented in Table 1. The K-pulse duration for this 12 meters long wire is known as 80 nsec ($T_k = 2 L / c$). As seen from Table 1, the optimized K-pulse duration is tabulated as 80.097 nsec that is in excellent agreement with the true value where the error is less than 0.1 percent. Also, to validate the pulse shape, spectrum zeros of the optimized K-pulse are computed from

$$K(s) = 0 \qquad (15)$$

where $K(s)$ is the Laplace transform of the K-pulse and s is the complex frequency. The resultant K-pulse spectrum zeros(i.e.,the estimated wire poles) are listed in Table 2 together with the known pole values of the wire revealing that the optimized K-pulse excitation has the ability of annihilating the wire's natural response not only at the synthesis aspect but also at all arbitrarily chosen aspect angles due to a proper pole-zero cancellation.

Table 1. Optimized parameters for the wire K-pulse

Optimization Variable	Optimization Interval	Optimized Value
T_k	[60 nsec, 100 nsec]	80.097 nsec
D_0	[-2, 2]	0.9893
D_1	[-2, 2]	-0.2202
D_2	[-2, 2]	0.1533
D_3	[-2, 2]	-0.1484
D_4	[-2, 2]	0.0645
D_5	[-2, 2]	-0.0547

Table 2. K-pulse spectrum zeros and system poles for the conducting wire (first six pole pairs only, in sL/c scale)

K-pulse Spectrum Zeros	Wire Poles (CNR frequencies)
-0.156 ± j 3.026	-0.161± j 3.013
-0.247 ± j 6.174	-0.219 ± j 6.133
-0.308 ± j 9.356	-0.262 ± j 9.256
-0.332 ± j 12.512	-0.294 ± j 12.383
-0.344 ± j 15.659	-0.320 ± j 15.512
-0.350 ± j 18.803	-0.343 ± j 18.642
⋮	⋮

The impulse response and the K-pulse response waveforms of the conducting wire are plotted in Figures 6, 7 and 8 at the aspect angles of 30, 60 and 90 degrees, respectively to support that conclusion. It is clearly observed in all these three cases that the K-pulse response is time-limited as expected.

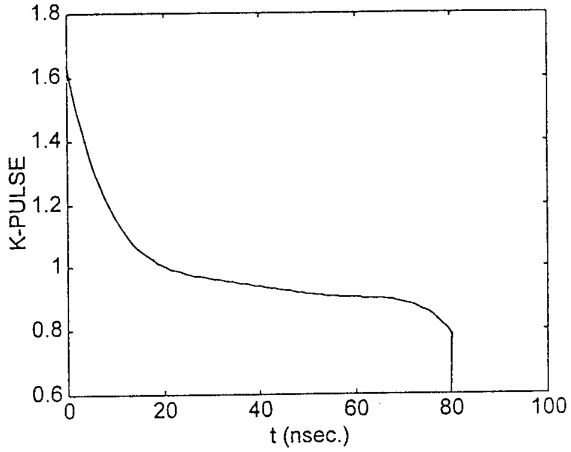


Figure 5. Optimized K-pulse for conducting thin wire

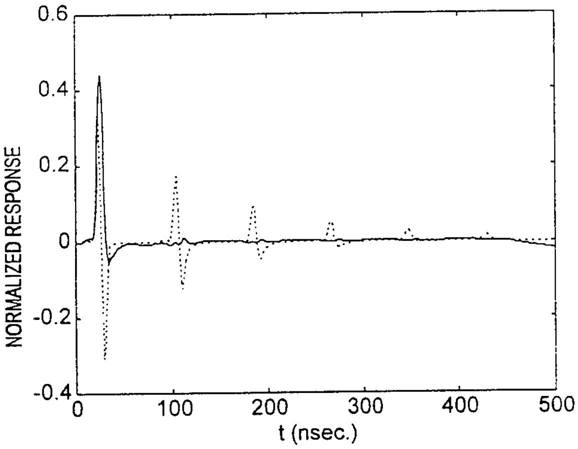


Figure 6. Scattered impulse response (...) and K-pulse response (—) of the wire at 30 degrees aspect angle measured from the wire axis.

A demonstration for target classification is also given in Figure 9 where the targets are three thin conducting wires of lengths 12 meters (wire W_1), 9 meters (wire W_2) and 6 meters (wire W_3), all having the same L/d ratio of 2000. The impulse response of W_1 at 90 degrees aspect angle is used as the test signal to be convolved with the K-pulse signals of the candidate targets. As shown in Figure 9, the only response with a finite support is created by the matched K-pulse filter, the outputs of the other K-pulse filters keep oscillating due to the natural response of the wire W_1.

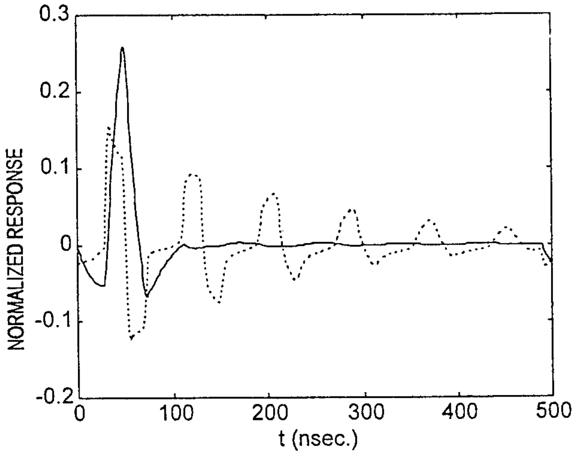


Figure 7. Scattered impulse response (...) and K-pulse response (—) of the wire at 60 degrees aspect angle measured from the wire axis.

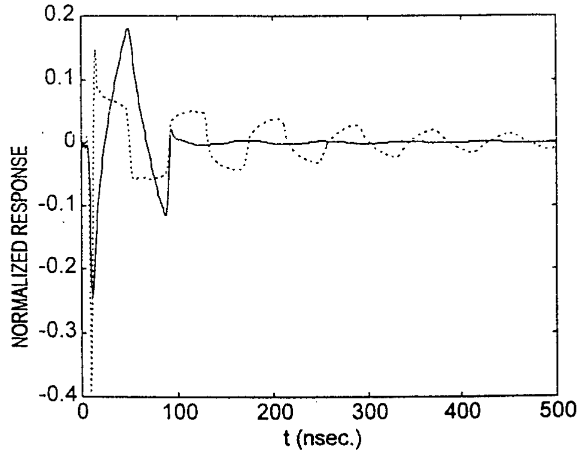


Figure 8. Scattered impulse response (...) and K-pulse response (—) of the wire at 90 degrees aspect angle measured from the wire axis.

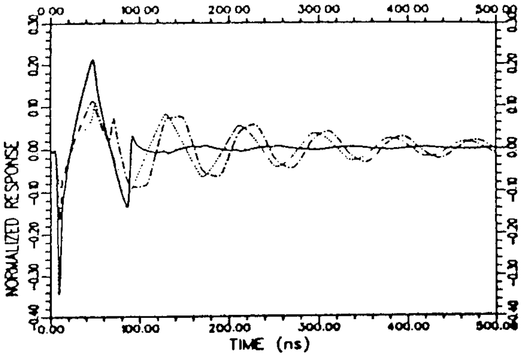


Figure 9. Response of the wire W_1 to its own K-pulse (—), to the K-pulse of the wire W_2 (-.-.-) and to the K-pulse of the wire W_3 (......) at 90 degrees aspect angle.

4. CONCLUSIONS

This paper has presented two electromagnetic target classification techniques which benefit from the learning, self organizing and evolutionary algorithms. The former technique makes use of artificial neural networks, the SOM network in particular, together with a novel feature extraction technique based on Wigner distributions. This type of target classifier needs a large scattered signal database providing information about the targets at many aspect angles and/or polarizations. Having data over a wide frequeny band brings an advantage but the technique is applicable in the presence of narrow band data as well. The latter target classification technique, on the other hand, is a natural-resonance based technique which needs quite a wide band data at only a few aspects and/or polarizations. The genetic algorithms make the synthesis of K-pulses possible in this technique especially when the K-pulse duration itself is an unknown.

In both of these target classification techniques, the correct classification rate is high (typically over % 90) and the computer cpu time needed for real-time classification is minimized down to fractions of a second. Research for further improvements in these target classification techniques is continuing and the results will be reported in future publications.

REFERENCES

1. I. Jouny, E.D. Garber and S.C. Ahalt, " Classification of radar targets using synthetic neural networks," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, No. 2, pp.336-343, April 1993.
2. S. Haykin and C. Deng, " Classification of radar clutter using neural networks," *IEEE Transactions on Neural Networks,* Vol. 2, No. 6, pp. 589-600, November 1991.

3. T. Ince, Electromagnetic Target Classification by Using Time-Frequency Analysis and Neural Networks, Master Thesis, Middle East Technical University, August 1996.
4. S. Haykın, Neural Networks: A Comprehensive Foundation, Macmillan Collage Publishing Company, New York , 1994.
5. T. Kohonen, Self-Organization and Associative Memory , Springer-Verlag, 1989
6. L. Cohen, Time-Frequency Analysis , Prentice Hall PTR, Prentice Hall Inc., NJ, 1995.
7. G.Turhan-Sayan, K.Leblebicioglu and T.Ince, "Electromagnetic Target Classification Using Time-Frequency Analysis and Neural Networks," submitted manuscript.
8. E. M. Kennaugh, "The K-Pulse Concept," *IEEE Trans. Antennas Propagat.*, Vol. AP-29, March 1981, pp. 327-331.
9. G. Turhan-Sayan and D. L. Moffatt, "K-Pulse Estimation and Target Identification of Low-Q Radar Targets," *Wave Motion*, No. 11, Nov. 1989, pp. 453-461.
10. G. Turhan-Sayan and D. L. Moffatt, "K-Pulse Estimation and Target Identification for Geometrically Complicated Low-Q Scatterers," in *Ultra-Wideband Radar: Proceedings of the First LosAlamos Symposium*, Edited by B.W. Noel, CRC Press, 1991, pp. 435-462.
11. F.Y.S. Fok, D.L.Moffatt and N. Wang, " K-Pulse Estimation From the Impulse Response of a Target," *IEEE Trans. Antennas Propagat.*, Vol. AP-35, Aug. 1987, pp. 926-933.
12. G.Turhan-Sayan, K.Leblebicioglu and S.Inan, " Input Signal Shaping for Target Identification Using Genetic Algorithms," *Microwave and Optical Technology Letters*, Vol.17, No.2, Feb. 1998, pp.128-132.
13. E.M. Kennaugh and D.L. Moffatt, "Transient and Impulse Response Approximations," *Proc. IEEE*, Vol. 53, Aug. 1965, pp. 893-901.
14. L. Marin, "Natural-Mode Representation of Transient Scattered Fields," *IEEE Trans. Antennas Propagat.*, Vol. AP-21, Nov. 1973, pp.809-818.
15. G. Turhan-Sayan and D. L. Moffatt, "K-Pulse Estimation Using Legendre Polynomial Expansions and Target Discrimination," *Journal of Electromagnetic Waves and Applications*, Vol. 4, No. 2, 1990, pp. 113-128.
16. D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.
17. L. Davis (ed.), *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, New York, 1991.
18. Weile and E. Michielssen, "Genetic Algorithm Optimization Applied to Electromagnetics: A Review," *IEEE Trans. Antennas Propagat.,* Vol. AP-45, March 1997, pp.343-353.

REPORT DOCUMENTATION PAGE

1. Recipient's Reference	**2. Originator's References** RTO MP-3 AC/323(SCI)TP/1	**3. Further Reference** ISBN 92-837-1006-1	**4. Security Classification of Document** UNCLASSIFIED/ UNLIMITED

5. Originator Research and Technology Organization
North Atlantic Treaty Organization
BP 25, 7 rue Ancelle, F-92201 Neuilly-sur-Seine Cedex, France

6. Title

The Application of Information Technologies (Computer Science) to Mission Systems

7. Presented at/sponsored by

The Symposium of the Systems Concepts and Integration Panel (SCI) held in Monterey, California, USA, 20-22 April 1998.

8. Author(s)/Editor(s) Multiple	**9. Date** November 1998
10. Author's/Editor's Address Multiple	**11. Pages** 254

12. Distribution Statement There are no restrictions on the distribution of this document. Information about the availability of this and other RTO unclassified publications is given on the back cover.

13. Keywords/Descriptors

Information technology
Mission effectiveness
Information systems
Command and control
Knowledge bases
Situation awareness
Assets
NATO
Defense economics
Education
Decision making
Planning
Computer architecture
Integrated systems
Computerized simulation
Systems engineering
Artificial intelligence

14. Abstract

This volume contains the Technical Evaluation Report, and the 21 unclassified papers, presented at the Symposium of the Systems Concepts and Integration Panel (SCI) held in Monterey, California, USA, 20-22 April 1998.

The papers presented covered the following headings:

- Information System Architecture
- Information Availability about Mission Situation
- Knowledge Availability
- Systems

NORTH ATLANTIC TREATY ORGANIZATION

RESEARCH AND TECHNOLOGY ORGANIZATION
BP 25 • 7 RUE ANCELLE
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • Télex 610 176

DIFFUSION DES PUBLICATIONS
RTO NON CLASSIFIEES

L'Organisation pour la recherche et la technologie de l'OTAN (RTO), détient un stock limité de certaines de ses publications récentes, ainsi que de celles de l'ancien AGARD (Groupe consultatif pour la recherche et les réalisations aérospatiales de l'OTAN). Celles-ci pourront éventuellement être obtenues sous forme de copie papier. Pour de plus amples renseignements concernant l'achat de ces ouvrages, adressez-vous par lettre ou par télécopie à l'adresse indiquée ci-dessus. Veuillez ne pas téléphoner.

Des exemplaires supplémentaires peuvent parfois être obtenus auprès des centres nationaux de distribution indiqués ci-dessous. Si vous souhaitez recevoir toutes les publications de la RTO, ou simplement celles qui concernent certains Panels, vous pouvez demander d'être inclus sur la liste d'envoi de l'un de ces centres.

Les publications de la RTO et de l'AGARD sont en vente auprès des agences de vente indiquées ci-dessous, sous forme de photocopie ou de microfiche. Certains originaux peuvent également être obtenus auprès de CASI.

CENTRES DE DIFFUSION NATIONAUX

ALLEMAGNE
Fachinformationszentrum Karlsruhe
D-76344 Eggenstein-Leopoldshafen 2

BELGIQUE
Coordinateur RTO - VSL/RTO
Etat-Major de la Force Aérienne
Quartier Reine Elisabeth
Rue d'Evere, B-1140 Bruxelles

CANADA
Directeur - Gestion de l'information
(Recherche et développement) - DRDGI 3
Ministère de la Défense nationale
Ottawa, Ontario K1A 0K2

DANEMARK
Danish Defence Research Establishment
Ryvangs Allé 1
P.O. Box 2715
DK-2100 Copenhagen Ø

ESPAGNE
INTA (RTO/AGARD Publications)
Carretera de Torrejón a Ajalvir, Pk.4
28850 Torrejón de Ardoz - Madrid

ETATS-UNIS
NASA Center for AeroSpace Information (CASI)
Parkway Center, 7121 Standard Drive
Hanover, MD 21076

FRANCE
O.N.E.R.A. (Direction)
29, Avenue de la Division Leclerc
92322 Châtillon Cedex

GRECE
Hellenic Air Force
Air War College
Scientific and Technical Library
Dekelia Air Force Base
Dekelia, Athens TGA 1010

ISLANDE
Director of Aviation
c/o Flugrad
Reykjavik

ITALIE
Aeronautica Militare
Ufficio Stralcio RTO/AGARD
Aeroporto Pratica di Mare
00040 Pomezia (Roma)

LUXEMBOURG
Voir Belgique

NORVEGE
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25
N-2007 Kjeller

PAYS-BAS
RTO Coordination Office
National Aerospace Laboratory NLR
P.O. Box 90502
1006 BM Amsterdam

PORTUGAL
Estado Maior da Força Aérea
SDFA - Centro de Documentação
Alfragide
P-2720 Amadora

ROYAUME-UNI
Defence Research Information Centre
Kentigern House
65 Brown Street
Glasgow G2 8EX

TURQUIE
Millî Savunma Başkanliği (MSB)
ARGE Dairesi Başkanliği (MSB)
06650 Bakanliklar - Ankara

AGENCES DE VENTE

NASA Center for AeroSpace Information (CASI)
Parkway Center
7121 Standard Drive
Hanover, MD 21076
Etats-Unis

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
Royaume-Uni

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council
Document Delivery,
Montreal Road, Building M-55
Ottawa K1A 0S2
Canada

Les demandes de documents RTO ou AGARD doivent comporter la dénomination "RTO" ou "AGARD" selon le cas, suivie du numéro de série (par exemple AGARD-AG-315). Des informations analogues, telles que le titre et la date de publication sont souhaitables. Des références bibliographiques complètes ainsi que des résumés des publications RTO et AGARD figurent dans les journaux suivants:

Scientific and Technical Aerospace Reports (STAR)
STAR peut être consulté en ligne au localisateur de ressources uniformes (URL) suivant:
http://www.sti.nasa.gov/Pubs/star/Star.html
STAR est édité par CASI dans le cadre du programme NASA d'information scientifique et technique (STI)
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
Etats-Unis

Government Reports Announcements & Index (GRA&I)
publié par le National Technical Information Service
Springfield
Virginia 2216
Etats-Unis
(accessible également en mode interactif dans la base de données bibliographiques en ligne du NTIS, et sur CD-ROM)

Imprimé par le Groupe Communication Canada Inc.
(membre de la Corporation St-Joseph)
45, boul. Sacré-Cœur, Hull (Québec), Canada K1A 0S7

NORTH ATLANTIC TREATY ORGANIZATION

RESEARCH AND TECHNOLOGY ORGANIZATION
BP 25 • 7 RUE ANCELLE
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Telefax 0(1)55.61.22.99 • Telex 610 176

DISTRIBUTION OF UNCLASSIFIED RTO PUBLICATIONS

NATO's Research and Technology Organization (RTO) holds limited quantities of some of its recent publications and those of the former AGARD (Advisory Group for Aerospace Research & Development of NATO), and these may be available for purchase in hard copy form. For more information, write or send a telefax to the address given above. **Please do not telephone.**

Further copies are sometimes available from the National Distribution Centres listed below. If you wish to receive all RTO publications, or just those relating to one or more specific RTO Panels, they may be willing to include you (or your organisation) in their distribution.

RTO and AGARD publications may be purchased from the Sales Agencies listed below, in photocopy or microfiche form. Original copies of some publications may be available from CASI.

NATIONAL DISTRIBUTION CENTRES

BELGIUM
Coordinateur RTO - VSL/RTO
Etat-Major de la Force Aérienne
Quartier Reine Elisabeth
Rue d'Evere, B-1140 Bruxelles

CANADA
Director Research & Development
Information Management - DRDIM 3
Dept of National Defence
Ottawa, Ontario K1A 0K2

DENMARK
Danish Defence Research Establishment
Ryvangs Allé 1
P.O. Box 2715
DK-2100 Copenhagen Ø

FRANCE
O.N.E.R.A. (Direction)
29 Avenue de la Division Leclerc
92322 Châtillon Cedex

GERMANY
Fachinformationszentrum Karlsruhe
D-76344 Eggenstein-Leopoldshafen 2

GREECE
Hellenic Air Force
Air War College
Scientific and Technical Library
Dekelia Air Force Base
Dekelia, Athens TGA 1010

ICELAND
Director of Aviation
c/o Flugrad
Reykjavik

ITALY
Aeronautica Militare
Ufficio Stralcio RTO/AGARD
Aeroporto Pratica di Mare
00040 Pomezia (Roma)

LUXEMBOURG
See Belgium

NETHERLANDS
RTO Coordination Office
National Aerospace Laboratory, NLR
P.O. Box 90502
1006 BM Amsterdam

NORWAY
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25
N-2007 Kjeller

PORTUGAL
Estado Maior da Força Aérea
SDFA - Centro de Documentação
Alfragide
P-2720 Amadora

SPAIN
INTA (RTO/AGARD Publications)
Carretera de Torrejón a Ajalvir, Pk.4
28850 Torrejón de Ardoz - Madrid

TURKEY
Millî Savunma Başkanliği (MSB)
ARGE Dairesi Başkanliği (MSB)
06650 Bakanliklar - Ankara

UNITED KINGDOM
Defence Research Information Centre
Kentigern House
65 Brown Street
Glasgow G2 8EX

UNITED STATES
NASA Center for AeroSpace Information (CASI)
Parkway Center, 7121 Standard Drive
Hanover, MD 21076

SALES AGENCIES

NASA Center for AeroSpace Information (CASI)
Parkway Center
7121 Standard Drive
Hanover, MD 21076
United States

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
United Kingdom

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council
Document Delivery,
Montreal Road, Building M-55
Ottawa K1A 0S2
Canada

Requests for RTO or AGARD documents should include the word 'RTO' or 'AGARD', as appropriate, followed by the serial number (for example AGARD-AG-315). Collateral information such as title and publication date is desirable. Full bibliographical references and abstracts of RTO and AGARD publications are given in the following journals:

Scientific and Technical Aerospace Reports (STAR)
STAR is available on-line at the following uniform resource locator:
http://www.sti.nasa.gov/Pubs/star/Star.html
STAR is published by CASI for the NASA Scientific and Technical Information (STI) Program
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
United States

Government Reports Announcements & Index (GRA&I)
published by the National Technical Information Service
Springfield
Virginia 22161
United States
(also available online in the NTIS Bibliographic Database or on CD-ROM)

Printed by Canada Communication Group Inc.
(A St. Joseph Corporation Company)
45 Sacré-Cœur Blvd., Hull (Québec), Canada K1A 0S7

ISBN 92-837-1006-1